

ZT 8802 and ZT 88CT02 Single Board V40 Computers

OPERATING MANUAL

For ZT 8802 Revision 0
ZT 88CT02 Revision 0

Reorder Part Number ZT M8802
March 17, 1994



1050 Southwood Drive
San Luis Obispo, CA 93401 USA
FAX (805) 541-5088
Telephone (805) 541-0488

ZIATECH WARRANTY

For Ziatech Board- and System-Level Computer Products

FIVE-YEAR LIMITED WARRANTY

Products manufactured by Ziatech Corporation are covered from the date of purchase by a five-year warranty against defects in materials, workmanship, and published specifications applicable to the date of manufacture. During the warranty period, Ziatech will repair or replace, solely at its option, defective units provided they are returned at customer expense to an authorized Ziatech repair facility. Products which have been subjected to misuse, abuse, neglect, alteration, or unauthorized repair, determined at the sole discretion of Ziatech, whether by accident or otherwise, are excluded from warranty. The warranty on fans and disk drives is limited to two years, the warranty on software is limited to one year, and the warranty on flat panel displays is limited to nine months from date of purchase. Other products and accessories not manufactured by Ziatech are limited to the warranty provided by the original manufacturer. Consumable items (fuses, disk media, batteries, etc.) are not covered by this warranty.

Ziatech may offer, where applicable and available, replacement products; otherwise, repairs requiring components, assemblies, and other purchased materials may be limited by market availability.

Ziatech assumes no liability resulting from changes to government regulations affecting use of materials, equipment, safety, and methods of repair. Ziatech may, at its discretion, offer replacement products.

The above warranty is in lieu of any other warranty, whether expressed, implied, or statutory, including, but not limited to, any warranty for fitness of purpose, merchantability, or freedom from infringement or the like, and any warranty otherwise arising out of any proposal, specifications, or sample. Ziatech neither assumes nor authorizes any person to assume for it any other liability. The liability of Ziatech under this warranty agreement is limited to a refund of the purchase price. In no event shall Ziatech be liable for loss of profits, use, incidental, consequential, or other damage, under this agreement.

LIFE SUPPORT POLICY

Ziatech products are not authorized for use as critical components in life support devices or systems without the express written approval of the president of Ziatech Corporation. As used herein:

1. Life support devices or systems are devices or systems which support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be expected to cause the failure of the life support device or system, affect its safety, or limit its effectiveness.

©1994 Ziatech Corporation

Turbo Debugger is a registered trademark of Borland International, Inc.
MULTIMODULE is a trademark of Intel Corporation
AT, IBM, IBM PC, and PC XT are registered trademarks of International Business Machines, Inc.
MS-DOS is a registered trademark of Microsoft Corporation
Generation 4 is a registered trademark of Opto 22
DEBUG/RT is a trademark of Paradigm Systems
STD ROM is a trademark of Ziatech Corporation
STD 32 is a registered trademark of Ziatech Corporation

CUSTOMER SUPPORT

If you have a technical question, please call Ziatech's Customer Support Service at one of the following numbers:

Corporate Headquarters: (805) 541-0488
(805) 541-5088 (FAX)

Eastern USA Sales Office: (610) 524-9070
(610) 524-9072 (FAX)

You can also use a modem to leave a message on the 24-hour Ziatech Bulletin Board Service (BBS) by calling (805) 541-8218. The BBS will also provide you with current Ziatech product revision and upgrade information.

If you have a sales question, please contact your local Ziatech Sales Representative.

PREFACE

This manual describes the operation and use of the ZT 8802 and ZT 88CT02 Single Board V40 Computers. The boards are functionally identical; however, the ZT 88CT02 consumes less power and operates over a wider temperature range. Refer to Appendix B for the particular specifications of each version of the board. Throughout the manual, ZT 8802 refers to both products unless specifically noted otherwise.

The following organizational outline describes the focus of each chapter in this manual. Section headings enclosed in boxes indicate the locations of labeled tabs, which are provided for quick access to the appropriate information.

I. **INTRODUCTION**

Chapter 1, "Introduction," offers an overview of the ZT 8802. It includes a product definition, a list of product features, a functional block diagram, and a brief description of each block. If you wish to compare the features of the ZT 8802 against the needs of a specific application, this chapter is especially useful.

II. **GETTING STARTED**

Chapter 2, "Getting Started," summarizes the information you need to begin using your ZT 8802. This includes system requirements, memory and I/O mapping, and installation instructions. The "USER'S REFERENCE" section contains more detailed discussions of the topics presented in this chapter.

III. | | |------------------| | USER'S REFERENCE | |------------------|

Chapter 3, "Theory of Operation," presents a detailed description of system level operation. This chapter covers memory, I/O organization, and interrupt structure, and it answers some commonly asked questions about the ZT 8802.

Chapter 4, "Processor Description (V40)," divides the V40 into functional blocks and presents a theory of operation for each. This chapter is most useful to those who are not familiar with the architecture of the V40/8088 series microprocessor.

Chapter 5, "Processor Configuration (V40)," describes the architecture of a V40 software programmable register set used to configure peripherals resident on the V40 for specific applications. These peripherals are described in greater detail in the following chapters.

Chapter 6, "Counter/Timers (V40)," describes the V40 Counter/Timer Control Unit, which includes three 16-bit counter/timers and is functionally equivalent to the 8254 Programmable Interval Timer. This chapter also includes register descriptions.

Chapter 7, "Interrupt Controller (V40)," describes the V40 Interrupt Control Unit, a programmable interface between interrupt generating peripherals and the CPU. This chapter also includes register descriptions.

Chapter 8, "DMA Controller (V40)," describes the V40 Direct Memory Access Control Unit, a programmable peripheral device used to allow temporary masters (other bus masters) access to the on-board system memory. This chapter also includes register descriptions.

Chapter 9, "Serial Communications (V40)," describes the V40 Serial Control Unit, a single serial channel that performs asynchronous serial communication between the V40 and a serial device external to the ZT 8802. This chapter also provides register descriptions and baud rate information.

Chapter 10, "Serial Communications (16C452)," describes the two RS-232-C serial ports on the ZT 8802, serial port signals, and register addresses.

Chapter 11, "Watchdog Timer," describes the function, configuration, and operation of the ZT 8802 watchdog timer, which is used to monitor ZT 8802 operation and take corrective action if the ZT 8802 fails to function as programmed.

Chapter 12, "SBX Expansion Module," contains a description and installation information for the optional SBX expansion module, which can be used to expand the I/O capabilities of the ZT 8802.

Chapter 13, "Parallel I/O Adapter (16C49)," discusses the 16C49 Parallel Interface Adapter (PIA) in detail. This chapter covers all information necessary to program and interface to the 16C49. It also describes the ZT 2226 24-Channel I/O Mounting Rack, ZT 2223 Industrial I/O Adapter Board, and ZT 2225 Industrial I/O Cable Adapter as they are used with the ZT 8802 or ZT 88CT02 single board computer.

Chapter 14, "Real-Time Clock/Calendar (DS 1202)," describes the function, configuration, and operation of the real-time clock/calendar. The real-time clock provides timekeeping features that are useful in many applications.

IV. | | |------------| | APPENDICES | |------------|

Appendix A, "Jumper Configurations," provides a detailed explanation of the options you can select through the use of jumper configurations.

Appendix B, "Specifications," contains the electrical, environmental, and mechanical specifications for the boards. Frontplane connector pinouts and cable drawings are also included.

Appendix C, "PIA System Setup Considerations," illustrates precautions you should take to prevent the latchup conditions that can occur with CMOS technology.

Appendix D, "Customer Support," offers a product revision history, technical support information, and the necessary instructions should you need to return the ZT 8802 for service.

Appendix E, "Glossary," defines important terms and acronyms used throughout the manual.

CONTENTS

I. INTRODUCTION

Chapter 1. INTRODUCTION	1-1
OVERVIEW	1-2
FEATURES OF THE ZT 8802	1-4
FUNCTIONAL BLOCKS	1-7
V40 (μ PD70208) Processor	1-7
Memory and I/O Addressing	1-7
Wait-State Generator	1-8
Direct Memory Access (DMA)	1-8
Optional Battery Backup	1-8
DC Power-Fail Detection	1-9
Real-Time Clock/Calendar	1-9
Serial Communications	1-9
Counter/Timers	1-10
Interrupts	1-10
16C49 Digital I/O Port	1-11
Clock Shutdown	1-11

II. GETTING STARTED

Chapter 2. GETTING STARTED	2-1
OVERVIEW	2-2
UNPACKING	2-2
WHAT'S IN THE BOX?	2-3
SYSTEM REQUIREMENTS	2-4
Physical Requirements	2-4
Power Requirements	2-4

Contents

Environmental Requirements	2-5
INSTALLING THE ZT 8802	2-6
Configuring the ZT 8802 for STD ROM	2-9
Configuring the ZT 8802 for DOS	2-12
MEMORY ADDRESSING	2-19
Memory Modes	2-28
I/O ADDRESSING	2-30
JUMPER OPTIONS	2-30

III. **USER'S REFERENCE**

Chapter 3. THEORY OF OPERATION	3-1
OVERVIEW	3-3
COMMONLY ASKED QUESTIONS	3-4
RELATIVE MICROPROCESSOR PERFORMANCE	3-8
STD BUS COMPATIBILITY	3-9
STD INTERFACE	3-10
STD Bus Memory Transfers	3-10
STD Bus I/O Transfers	3-10
DIRECT MEMORY ACCESS (DMA)	3-12
Advantages of DMA	3-12
DMA Operation	3-13
I/O POINTS	3-15
Input/Output Operation	3-16
Masking Operation	3-18
Event Sense Operation	3-19
Electrical Specifications	3-21
Reset Operation	3-21
Power Supply Lines	3-22
Connectors	3-22
SERIAL COMMUNICATIONS	3-23
INTERRUPTS	3-24
Interrupt Request Assignments	3-24
Polled Interrupts on the STD Bus	3-26
STD Bus Vectored Interrupts	3-28
STD Bus Cascaded Interrupts	3-29
Non-Maskable Interrupts	3-30
RESET	3-31
CONTROL PORT	3-31

Hardwiring Features vs. Software Control	3-33
POWER-FAIL PROTECTION	3-34
DC Power Fail	3-34
BATTERY	3-35
STATUS INDICATOR (LED)	3-37
CMOS VERSION OF THE ZT 8802	3-37
Added Features	3-37

Chapter 4. PROCESSOR DESCRIPTION (V40) 4-1

OVERVIEW	4-2
ZT 8802 Specifics	4-3
FUNCTIONAL BLOCKS	4-4
CPU - Central Processing Unit	4-6
BIU - Bus Interface Unit	4-22
BAU - Bus Arbitration Unit	4-23
CGU - Clock Generator Unit	4-23
VCR - V40 Configuration Registers	4-24
WCU - Wait Control Unit	4-24
SCU - Serial Control Unit	4-24
TCU - Counter/Timer Control Unit	4-25
ICU - Interrupt Control Unit	4-25
DCU - DMA Control Unit	4-25
RESET	4-26
MEMORY AND I/O ADDRESSING	4-27
INTERRUPTS	4-30
Divide Error	4-33
Single-Step	4-34
Non-Maskable	4-34
Fixed Vector Instruction	4-34
Overflow	4-35
Check Index	4-35
Variable Vector Instruction	4-35
Emulation Mode	4-35
8080 EMULATION	4-36

Chapter 5. PROCESSOR CONFIGURATION (V40) 5-1

OVERVIEW	5-2
VCR - V40 CONFIGURATION REGISTERS	5-2
OPCN - On-Chip Peripheral Connection Register	5-4
OPSEL - On-Chip Peripheral Selection Register	5-5
OPHA, DULA, IULA, TULA, and SULA	5-6

Contents

WCY2 - Wait Cycle 2 Register	5-8
WCY1 - Wait Cycle 1 Register	5-8
WMB - Wait Memory Boundary Register	5-10
RFC - Refresh Control Register	5-11
TCKS - Counter/Timer Clock Selection Register	5-12
RESET	5-14
Chapter 6. COUNTER/TIMERS (V40)	6-1
OVERVIEW	6-2
ZT 8802 SPECIFICS	6-3
PROGRAMMABLE REGISTERS	6-5
Timer Mode Register (TMD)	6-6
Count Registers	6-10
Status Registers	6-10
ADDITIONAL INFORMATION	6-11
Chapter 7. INTERRUPT CONTROLLER (V40)	7-1
OVERVIEW	7-2
ZT 8802 SPECIFICS	7-3
Cascade Mode	7-5
PROGRAMMABLE REGISTERS	7-6
Initialization Words (IIW1, IIW2, IIW3, and IIW4)	7-7
Operation Words (IMKW, IPFW, and IMDW)	7-11
Status Words (IRQ, IIS, and IPOL)	7-15
ADDITIONAL INFORMATION	7-18
Chapter 8. DMA CONTROLLER (V40)	8-1
OVERVIEW	8-2
ZT 8802 SPECIFICS	8-3
Bus Request/Bus Acknowledge	8-3
PROGRAMMABLE REGISTERS	8-4
DMA Initialize Command (DICM)	8-5
DMA Channel (DCH)	8-5
DMA Base Count/Current Count (DBC/DCC)	8-7
DMA Base Address/Current Addr. (DBA/DCA)	8-8
DMA Device Control (DDC)	8-9
DMA Mode (DMD)	8-10
DMA Status (DST)	8-11
DMA Mask (DMK)	8-12
ADDITIONAL INFORMATION	8-12

Chapter 9. SERIAL COMMUNICATIONS (V40)	9-1
OVERVIEW	9-2
ZT 8802 SPECIFICS	9-3
RS-232-C Operation	9-3
PROGRAMMABLE REGISTERS	9-4
Serial Status Register (SST)	9-5
Serial Command Register (SCM)	9-6
Serial Mode Register (SMD)	9-7
Serial Interrupt Mask Register (SIMK)	9-8
BAUD RATE	9-9
ADDITIONAL INFORMATION	9-11
 Chapter 10. SERIAL COMMUNICATIONS (16C452)	 10-1
OVERVIEW	10-2
SERIAL COMMUNICATIONS PROTOCOL	10-3
SERIAL INTERFACE (RS-232-C)	10-7
Signal Definitions	10-9
SERIAL REGISTERS	10-14
Transmit and Receive Buffer Registers	10-18
Scratchpad Register	10-18
Line Control Register	10-19
Baud Rate Generator	10-22
Line Status Register	10-25
Interrupt ID Register	10-27
Interrupt Enable Register	10-29
Modem Control Register	10-30
Modem Status Register	10-32
 Chapter 11. WATCHDOG TIMER	 11-1
OVERVIEW	11-1
ZT 8802 SPECIFICS	11-2
FUNCTIONAL DESCRIPTION	11-2
Stage Timer	11-3
Stage Delay	11-3
OPERATION	11-4
Reset	11-4
Programming	11-5

Contents

Chapter 12. SBX EXPANSION MODULE	12-1
OVERVIEW	12-2
Features	12-2
ZT 8802 SPECIFICS	12-3
INSTALLATION	12-4
 Chapter 13. PARALLEL I/O ADAPTER (16C49)	 13-1
OVERVIEW	13-2
FEATURES OF THE 16C49	13-3
PROCESSOR INTERFACE/CONNECTOR PINOUT	13-4
Control Port	13-5
I/O INTERFACE	13-7
I/O Circuit Operation	13-7
I/O Circuit Specifications	13-9
Reset Operation	13-9
I/O Port Description	13-10
I/O Module Mounting Racks	13-14
Centronics Printer Interface	13-18
 Chapter 14. REAL-TIME CLOCK/CALENDAR (DS 1202)	 14-1
OVERVIEW	14-2
ZT 8802 SPECIFICS	14-2
FUNCTIONAL DESCRIPTION	14-3
OPERATION	14-3
COMMAND BYTE	14-5
Byte Mode	14-5
Burst Mode	14-6
Data Input/Output	14-7
Clock/Calendar	14-8
 IV. APPENDICES	
 Appendix A. JUMPER CONFIGURATIONS	 A-1
OVERVIEW	A-1
JUMPER OPTIONS	A-3

Appendix B. SPECIFICATIONS	B-1
ELECTRICAL AND ENVIRONMENTAL	B-2
Absolute Maximum Ratings	B-2
DC Operating Characteristics	B-2
Digital I/O Interface Characteristics	B-2
Battery Backup Characteristics	B-3
STD Bus Loading Characteristics	B-3
MECHANICAL	B-6
Card Dimensions & Weight	B-6
Connectors	B-9
Cables	B-17
Assembly Diagram	B-21
Appendix C. PIA SYSTEM SETUP CONSIDERATIONS	C-1
OVERVIEW	C-1
PREVENTING SYSTEM LATCHUP	C-2
POWER SUPPLY SEQUENCE MISMATCH	C-2
SIGNAL LEVEL MISMATCH	C-5
Appendix D. CUSTOMER SUPPORT	D-1
OVERVIEW	D-1
REVISION HISTORY	D-2
TECHNICAL ASSISTANCE	D-2
RELIABILITY	D-3
RETURNING FOR SERVICE	D-4
ZIATECH 5+5 WARRANTY	D-5
Appendix E. GLOSSARY	E-1
OVERVIEW	E-1
TERMINOLOGY	E-1

V. **INDEX**

Index	i
--------------------	----------

TABLES

Table 3-1	Processor Speed Comparison.	3-8
Table 3-2	Mask Port.	3-18
Table 3-3	Event Sense Port.	3-20
Table 3-4	I/O Line Electrical Specifications.	3-21
Table 3-5	STD Bus Interrupts.	3-24
Table 4-1	Segment Registers.	4-8
Table 4-2	Implied Use of General Registers.	4-11
Table 4-3	CPU Reset State.	4-26
Table 4-4	Interrupt Sources.	4-30
Table 4-5	Interrupt Vector Table.	4-32
Table 4-6	Emulation Mode Registers and Flags.	4-38
Table 5-1	V40 Configuration Registers.	5-3
Table 5-2	STD ROM and Ziotech DOS Address Selection.	5-7
Table 5-3	V40 Configuration Register Defaults after RESET.	5-14
Table 5-4	STD ROM V40 Configuration Register Defaults.	5-15
Table 5-5	DOS System V40 Configuration Register Defaults.	5-16
Table 6-1	TCU Register Addressing.	6-5
Table 7-1	Interrupt Controller Inputs.	7-3
Table 7-2	ICU Register Addressing.	7-6
Table 8-1	DCU Register Addressing.	8-4
Table 9-1	SCU Register Addressing.	9-4
Table 9-2	ZT 8802 Baud Rate Counts.	9-10
Table 10-1	16C452 Reset State.	10-12
Table 10-2	ZT 8802 I/O Port Assignments.	10-15
Table 10-3	16C452 Addressable Registers Summary.	10-16
Table 10-4	Baud Rates, 1.8432 MHz Clock.	10-23
Table 10-5	Baud Rates, 8 MHz Clock.	10-24
Table 10-6	16C452 Interrupt Control Functions.	10-27
Table 11-1	Watchdog Timer Stage Delay Options.	11-3
Table 13-1	16C49 PIA I/O Ports.	13-4
Table 13-2	I/O Point Specifications.	13-9
Table 13-3	I/O Point Write/Read Ports.	13-10
Table 13-4	Event Sense Port.	13-11

Tables

Table 13-5	Mask Port.	13-13
Table 13-6	ZT 8802 J6 Connection to ZT 2225.	13-17
Table A-1	Jumper Cross Reference.	A-2
Table A-2	Jumper Descriptions.	A-3
Table B-1	Bus Signal Loading, P Connector.	B-4
Table B-2	Bus Signal Loading, E Connector.	B-5
Table B-3	J1 Pin Assignments.	B-11
Table B-4	J2 Pin Assignments.	B-12
Table B-5	J3 and J4 Pin Assignments.	B-13
Table B-6	J6 Parallel Port Pinout.	B-14
Table B-7	J7 Pin Assignments.	B-15
Table B-8	J8 SBX Expansion Module Pinout.	B-16

ILLUSTRATIONS

Figure 1–1	ZT 8802 Functional Block Diagram.	1-6
Figure 2–1	STD ROM (32K ROM/128K RAM) Configuration.	2-8
Figure 2–2	DOS (128K Flash/256K RAM) Default.	2-15
Figure 2–3	DOS (512K ROM/1M RAM) Configuration.	2-16
Figure 2–4	512K ROM/1M RAM System Memory Map.	2-20
Figure 2–5	512K ROM/512K RAM System Memory Map.	2-21
Figure 2–6	512K ROM/256K RAM System Memory Map.	2-22
Figure 2–7	512K ROM/128K RAM System Memory Map.	2-23
Figure 2–8	128K/256K ROM-1M RAM Memory Map.	2-24
Figure 2–9	128K/256K ROM-512K RAM Memory Map.	2-25
Figure 2–10	128K/256K ROM-256K RAM Memory Map.	2-26
Figure 2–11	128K/256K ROM-128K RAM Memory Map.	2-27
Figure 2–12	I/O Map.	2-31
Figure 3–1	DMA With STD Bus Controller.	3-13
Figure 3–2	Typical I/O Circuit.	3-15
Figure 3–3	Typical Event Sense Circuit.	3-19
Figure 3–4	PIC Interrupt Input Requests.	3-25
Figure 3–5	Polled Interrupt Structure.	3-26
Figure 3–6	Small Scale Vectored Structure.	3-28
Figure 3–7	Cascaded Interrupt Structure.	3-29
Figure 3–8	Control Port Bit Map.	3-32
Figure 3–9	I/O Point Jumper Options.	3-33
Figure 4–1	V40 Block Diagram.	4-5
Figure 4–2	CPU Block Diagram.	4-7
Figure 4–3	Processor Status Word.	4-15
Figure 4–4	RESET and READY Synchronization.	4-22
Figure 4–5	Generic V40 Memory Map.	4-27
Figure 4–6	Data Formats.	4-28
Figure 4–7	Generic V40 I/O Map.	4-29
Figure 4–8	Interrupt Processing.	4-31
Figure 5–1	On-Chip Peripheral Connection Register.	5-4
Figure 5–2	On-Chip Peripheral Selection Register.	5-5
Figure 5–3	Wait-Cycle 2 Register.	5-8

Illustrations

Figure 5-4	Wait-Cycle 1 Register.	5-9
Figure 5-5	Wait-Cycle Memory Boundary Register.	5-10
Figure 5-6	Refresh Control Register.	5-11
Figure 5-7	Counter/Timer Clock Selection Register.	5-12
Figure 6-1	J2 Counter/Timer 2 Connector.	6-4
Figure 6-2	Counter/Timer General Mode Register.	6-7
Figure 6-3	Counter/Timer Count Mode Register.	6-8
Figure 6-4	Counter/Timer Multiple Mode Register.	6-9
Figure 6-5	Counter/Timer Count Register.	6-10
Figure 6-6	Counter/Timer Status Register.	6-11
Figure 7-1	Interrupt Jumper Selection.	7-4
Figure 7-2	Alternate IR4/IR3 Jumper Selections.	7-5
Figure 7-3	Interrupt Initialization Programming.	7-7
Figure 7-4	Interrupt Initialization Word 1.	7-8
Figure 7-5	Interrupt Initialization Word 2.	7-9
Figure 7-6	Interrupt Initialization Word 3.	7-9
Figure 7-7	Interrupt Initialization Word 4.	7-10
Figure 7-8	Interrupt Mask Word.	7-11
Figure 7-9	Interrupt Priority and Finish Word.	7-12
Figure 7-10	Interrupt Mode Word.	7-14
Figure 7-11	Interrupt Status Registers IRQ and IIS.	7-16
Figure 7-12	Interrupt Status Register IPOL.	7-17
Figure 8-1	DMA Initialization Command Register.	8-5
Figure 8-2	DMA Channel Register.	8-6
Figure 8-3	DMA Base & Current Count Registers.	8-7
Figure 8-4	DMA Base & Current Address Registers.	8-8
Figure 8-5	DMA Device Control Registers.	8-9
Figure 8-6	DMA Mode Register.	8-10
Figure 8-7	DMA Status Register.	8-11
Figure 8-8	DMA Mask Register.	8-12
Figure 9-1	J7 RS-232 DCE Configuration.	9-3
Figure 9-2	Serial Status Register.	9-5
Figure 9-3	Serial Command Register.	9-6
Figure 9-4	Serial Mode Register.	9-7
Figure 9-5	Serial Interrupt Mask Register.	9-8
Figure 10-1	Establishing Serial Communications.	10-4
Figure 10-2	16C452 Serial Port Block Diagram.	10-8
Figure 11-1	Watchdog Timer Functional Diagram.	11-2
Figure 12-1	SBX Expansion Module Installation.	12-4
Figure 13-1	16C49 PIA Block Diagram.	13-2
Figure 13-2	Control Port Bit Map.	13-6

Figure 13-3	Typical I/O Circuit.	13-7
Figure 13-4	Event Sense Inputs.	13-12
Figure 14-1	Command Byte Bit Format (Byte Mode).	14-5
Figure 14-2	Command Byte Bit Format (Burst Mode).	14-6
Figure 14-3	Clock/Calendar Registers.	14-9
Figure A-1	ZT 8802 Interrupt Jumper Selection.	A-8
Figure A-2	Alternate IR4/IR3 Jumper Selection.	A-9
Figure A-3	Control Port Configuration.	A-11
Figure A-4	Customer Jumper Configuration.	A-16
Figure A-5	DOS (128K Flash/256K RAM) Default.	A-17
Figure A-6	DOS (512K ROM/1M RAM) Configuration.	A-18
Figure A-7	STD ROM (32K ROM/128K RAM) Configuration. ...	A-19
Figure B-1	ZT 8802 Card Dimensions.	B-7
Figure B-2	Connector Locations.	B-8
Figure B-3	P/E Connector Pinout.	B-10
Figure B-4	ZT 90069 Revision B Serial Cable.	B-17
Figure B-5	ZT 90089 Revision A Digital I/O Cable.	B-18
Figure B-6	ZT 90136 Revision 0 Serial Cable.	B-19
Figure B-7	ZT 90156 Revision 0 Printer Interface Cable.	B-20
Figure B-8	ZT 8802 Assembly Diagram.	B-21
Figure C-1	I/O Rack Vcc and Ground Supplied Via Interface Cable. ...	C-2
Figure C-2	I/O Rack Vcc and Ground Supplied Externally.	C-3
Figure C-3	Computer-Switched External Power Supply.	C-3
Figure C-4	Computer & Ext. Power Supply w/ Common Switch. ...	C-4
Figure C-5	I/O Rack Vcc Supplied Externally, Common Ground. ...	C-5
Figure C-6	Computer-Switched External PS, Common Ground.	C-6
Figure C-7	Computer & External PS w/ Common Switch & Ground. ...	C-6

INTRODUCTION

Contents	Page
OVERVIEW	1-2
FEATURES OF THE ZT 8802	1-4
FUNCTIONAL BLOCKS	1-7
V40 (μ PD70208) Processor	1-7
Memory and I/O Addressing	1-7
Wait-State Generator	1-8
Direct Memory Access (DMA)	1-8
Optional Battery Backup	1-8
DC Power-Fail Detection	1-9
Real-Time Clock/Calendar	1-9
Serial Communications	1-9
Counter/Timers	1-10
Interrupts	1-10
16C49 Digital I/O Port	1-11
Clock Shutdown	1-11

OVERVIEW

The ZT 8802 is an 8 MHz, 8-bit, Single Board Computer (SBC) designed for control applications on the STD bus. Ziatech offers DOS (MS-DOS®) and STD ROM™ (Borland Turbo Debugger® environment using Paradigm Systems' DEBUG/RT™) as development platforms on the ZT 8802. The ZT 8802's high level of integration allows for a complete DOS system on one board. On-board PC peripherals are located at the same I/O addresses as on the IBM PC®, which provides a greater degree of software compatibility.

The ZT 88CT02 is the CMOS version of the ZT 8802. It is designed for an extended temperature range and low power applications. All references to the ZT 8802 in this manual also apply to the ZT 88CT02, unless otherwise noted.

The use of the NEC V40 microprocessor, an 8088 compatible processor with a superset of the 8088 instruction set, helps increase performance over that of 8088-based STD CPU boards. The V40 is a CMOS device, which results in lower power consumption. To further reduce power consumption, the ZT 8802 utilizes CMOS technology for most of the peripheral logic. The ZT 88CT02 is composed entirely of TTL compatible, CMOS devices.

Peripherals include three timer/counters, an interrupt controller, a real-time clock, three RS-232-C serial ports, a 48-line digital I/O interface, and three 32-pin memory sockets. One memory socket accepts a Flash or EPROM device of up to 512 Kbytes, and each of the other two memory sockets accepts up to 512 Kbytes of RAM. A bank switching mechanism is employed to permit up to 1 Mbyte of RAM and 512 Kbytes of EPROM all within the 1 Mbyte addressing range of the V40. A general purpose LED indicator and pushbutton reset are also provided.

All RAM and the real-time clock may be optionally battery-backed by a 1 Amp-hour lithium battery. DC power failure detection is provided to switch to the battery backup mode during +5 VDC failure.

Ziatech's DOS and STD ROM options provide software support. DOS includes the MS-DOS operating system for the ZT 8802. STD ROM provides software development capabilities when used in conjunction with an IBM PC. The STD ROM option is useful for applications in which the target system does not require an operating system.

FEATURES OF THE ZT 8802

- STD-80 bus compatible
- Optional CMOS version available (ZT 88CT02)
- 8088/8086 code compatible
- Three 32-pin memory sockets, configurable for two RAMs and one EPROM/Flash
- Acceptable RAM sizes are 32 Kbytes through 512 Kbytes
- Acceptable EPROM/Flash sizes are 32 Kbytes through 512 Kbytes
- +5 V-only operation (+12 V required for Flash programming)
- One 8259 compatible interrupt controller (V40)
- Backplane cascaded interrupt support
- Three RS-232-C serial channels (V40/16C452)
- Three 8254 compatible counter/timers (V40)
- 48-point Opto 22 compatible digital I/O interface (16C49)
- Eight points of event sense
- SBX expansion module interface

- Latching frontplane connectors
- Real-time clock/calendar (DS 1202)
- Optional battery backup for all RAM and clock
- Watchdog timer
- DC power-fail protection
- Programmable LED indicator
- Pushbutton reset switch
- Optional DOS operating system software
- Optional STD ROM development/debug software
- Optional industrial I/O adapter board (ZT 2223), cable adapter (ZT 2225), and 24-channel I/O module mounting rack (ZT 2226)
- Optional cables for serial port, digital I/O port, and Centronics printer interface
- Fully tested while cycling temperature from ambient to 55° C to guarantee reliability (to 80° C for ZT 88CT02)
- Sleep mode provided for on-board oscillators
- Five-year warranty

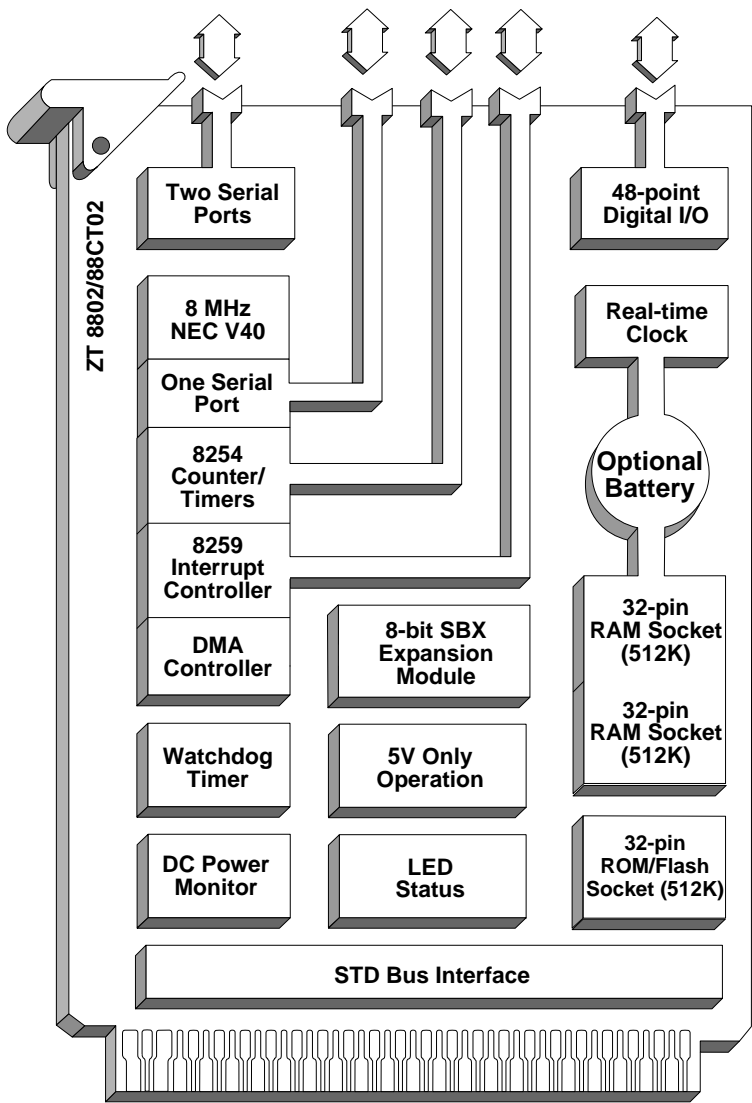


Figure 1-1. ZT 8802 Functional Block Diagram.

FUNCTIONAL BLOCKS

Figure 1-1 illustrates the ZT 8802's functional blocks. A brief description of each block follows.

V40 (μ PD70208) Processor

The NEC V40 is an 8088 compatible microprocessor with a 16-bit internal data bus and an 8-bit external data bus. The V40 executes all code written for the 8088/8086 family of microprocessors and includes a superset of their instruction set. Performance enhancements are provided by way of such architectural features as a dual 16-bit internal data bus, high-speed effective address generation, and additional hidden temporary registers. The added instructions include shift and rotate by immediate value, move string, stack manipulations, and 8080 emulation mode. The 8080 emulation mode enables existing 8-bit 8080 software to run on new 16-bit hardware with few or no software modifications.

Memory and I/O Addressing

The ZT 8802 has three 32-pin JEDEC compatible byte-wide sockets. One of the sockets accepts 128 through 512 Kbyte EPROM/Flash; the other two accept 32 through 512 Kbyte RAMs. The EPROM/Flash socket supports both 5 V and 12 V Flash memory.

20-bit addressing is used for on-board accesses, giving the system 1 Mbyte of direct address space. STD bus accesses have an additional four bits (A20-23) driven low to be compatible with 24-bit addressed memory boards. I/O accesses are accomplished with a 16-bit address, providing 64 Kbytes of I/O space for 16-bit addressed I/O boards or 256 bytes for 8-bit addressed boards.

Introduction

Wait-State Generator

The ZT 8802 contains a wait-state generator to accommodate I/O and memory boards that need more access time. If enabled, the wait-state generator inserts up to three clock cycles within the normal four clock bus cycle to increase the cycle to seven clocks. This gives memory and I/O boards additional time between address valid time and the end of the bus cycle to complete an access. DOS and STD ROM program one additional wait state (five clock transfer) by default.

Direct Memory Access (DMA)

The ZT 8802 supports external DMA controllers via the BUSRQ* (pin 42) and BUSAK* (pin 41) STD bus control signals. A request for the bus is made to the ZT 8802 via BUSRQ*, and the ZT 8802 responds with BUSAK* once the microprocessor has signaled its release of the bus. When the DMA transfer is complete, the DMA device releases BUSRQ* and the ZT 8802 then responds by releasing BUSAK*. The ZT 8802 supports DMA for all on-board EPROM and RAM. An example of an external DMA controller is the ZT 8950; it may be used to service an STD I/O device with DMA capability to and from the ZT 8802 memory.

Optional Battery Backup

All RAM and the real-time clock may be selectively battery-backed with a 3.6 V, 1 Amp-hour lithium battery, shipped as a standard option with ZT 8802 DOS systems. When DC power falls below 4.75 V, the battery power switches in and remains until power is again at 4.75 V. At the same time, the DCPDN* STD bus signal (pin 6) is driven active (low) to warn other boards in the system of low DC voltage.

DC Power-Fail Detection

DC power-fail detection senses when DC voltage drops below 4.75 V. This signals the board to switch into battery backup mode, as described above.

Real-Time Clock/Calendar

The real-time clock/calendar on the ZT 8802 is a Dallas Semiconductor DS 1202. It keeps track of seconds, minutes, hours, days, day of the month, month, and year. The clock automatically corrects for leap years and adjusts for months with fewer than 31 days. The optional battery will back up the real-time clock/calendar.

Serial Communications

The ZT 8802 contains three serial channels. One of the serial channels is provided by the V40 and is 8251 compatible. The other two serial channels are provided by the 16C452 and are 16C450 compatible (DOS COM compatible). All three channels are available as RS-232-C. The RS-232-C drivers make use of a charge pump so that only +5 V is required.

Introduction

Counter/Timers

The ZT 8802 has three independent 16-bit counter/timers, each of which can be used as a timer or event counter. The clock frequency driving each of these timers is a 1.19318 MHz oscillator. An optional off-board source can drive the counter/timers via J2.

The six programmable counter/timer modes are as follows:

1. Interrupt on end of count
2. Frequency divider
3. Square wave generator
4. Software-triggered strobe
5. Retriggerable hardware-triggered strobe
6. Retriggerable one-shot

The output of timer 2 is available at connector J2. The "gate" or enable input to timer 2 is pulled up active by a 100 k Ω resistor and may be controlled by a source on frontplane connector J2.

Interrupts

The programmable interrupt controller (PIC) on the ZT 8802 is equivalent to an Intel 8259A. It has eight interrupt inputs that can be prioritized in software. Its output drives the CPU interrupt input. All PIC interrupt inputs may be jumper selected between various on-board sources and the five frontplane and four backplane sources. Factory default assigns the DOS compatible interrupt selections as described by the jumper descriptions in Appendix A (see page A-7).

The interrupt structure follows Version 2.3 and later of the *STD-80 Bus Specification and Designer's Guide*, which allows for the RESERVED and CNTRL* signals (STD bus pins 37 and 50, respectively) to be interrupt sources as well as INTRQ* (pin 44). These signals are now referred to as INTRQ1*, INTRQ2*, and INTRQ*, respectively. The ZT 8802 also supports INTRQ3* as defined by the *STD 32 Bus Specification*. This interrupt structure provides for more backplane interrupts and may eliminate frontplane cabling for additional interrupts.

The STD-80 bus protocol for PIC cascading is also supported, allowing for 8259A interrupt controller expansion. The PIC may handle up to 57 prioritized interrupts by combining seven off-board sources, each of which may support eight interrupt inputs via a separate "slave" interrupt controller, plus one direct on-board source (IR0 is available only on board).

16C49 Digital I/O Port

The ZT 8802 provides up to 48 lines of digital I/O via the 16C49 event sense peripheral interface. Eight of these lines are used for on-board configuration, but may be configured for off-board I/O at the expense of some on-board features. The on-board features controlled by the 16C49 include the LED, memory page switching, sleep mode, on-board real-time clock, watchdog strobe, and optional control of a daughter board 5-to-12-V adapter for +12 V Flash devices.

Clock Shutdown

For power conservation, the ZT 8802 can shut down the SBX and timer-tick oscillators to conserve power.

GETTING STARTED

Contents	Page
OVERVIEW	2-2
UNPACKING	2-2
WHAT'S IN THE BOX?	2-3
SYSTEM REQUIREMENTS	2-4
Physical Requirements	2-4
Power Requirements	2-4
Environmental Requirements	2-5
INSTALLING THE ZT 8802	2-6
Configuring the ZT 8802 for STD ROM	2-9
STD ROM Memory Requirements	2-9
STD ROM Cable Requirements	2-10
STD ROM Jumper Configuration	2-10
STD ROM Operation	2-10
Configuring the ZT 8802 for DOS	2-12
DOS Memory Requirements	2-12
DOS Cable Requirements	2-13
DOS Jumper Configuration	2-14
Powering Up The DOS System	2-17
MEMORY ADDRESSING	2-19
Memory Modes	2-28
Mode Hardwiring	2-29
I/O ADDRESSING	2-30
JUMPER OPTIONS	2-30

OVERVIEW

This chapter includes all the information you need to properly install the ZT 8802 into an STD bus card cage. You should read this chapter and Chapter 3, "Theory of Operation," before you attempt to use the board. Remember, unless specifically stated otherwise, all references to the ZT 8802 also pertain to the ZT 88CT02.

UNPACKING

Please check the shipping carton for damage. If the shipping carton and contents are damaged, notify the carrier and Ziatech for an insurance settlement. Retain the shipping carton and packing material for inspection by the carrier. Do not return any product to Ziatech without a Return Material Authorization (RMA) number. Appendix D explains the procedure you should follow to obtain an RMA number from Ziatech.

WHAT'S IN THE BOX?

The items listed below are included in a standard ZT 8802 order. The list does not include options such as system level software or cabling. Refer to the packing list for a complete list of items shipped. When ordering specific system level software options with the ZT 8802, refer to the software manual for a list of the items that should be included.

- ZT 8802 V40 Single Board Computer or ZT 88CT02 V40 Single Board Computer
- Optional ZT 8802 Operating Manual (in binder)
- Anti-static packing material

Attach the sticker packaged with the manual to the spine of the binder. Be sure to save the anti-static packing material for use in storing or shipping the ZT 8802.

WARNING!

Like all equipment utilizing CMOS devices, the ZT 8802 must be protected from static discharge. This is especially important for the ZT 88CT02, which contains all CMOS logic and is therefore very sensitive to static discharge. Never remove or install any of the socketed parts except at a static-free workstation.

SYSTEM REQUIREMENTS

Physical Requirements

The ZT 8802 is designed to be used in an STD bus system. Therefore, it is physically and electrically compatible with the STD-80 bus standard. The board should normally be mounted in one slot of an STD bus card cage. Refer to the board outline on page B-7 for board dimensions.

Power Requirements

Power requirements for the ZT 8802 and ZT 88CT02 are +5 VDC at 900 mA maximum, 550 mA typical. ± 12 V is not needed for either board. The serial drivers use a charge pump mechanism to supply RS-232-C compatible signal levels. +12 V is required if +12 V Flash memory is used or if an expansion module daughter board requires +12 V.

WARNING!

If you are using an emulator in place of the microprocessor on the ZT 8802, the emulator should be powered down or disconnected before the STD card cage is powered down.

Environmental Requirements

The ambient temperature must be maintained at 0° to 65° Celsius for proper operation and to avoid possible damage to the ZT 8802 (the ZT 88CT02 allows for a lower power requirement and a wider temperature range, as detailed in Appendix B). Relative humidity should be less than 95% at 40° C, noncondensing.

We recommend you install the board vertically if you are using it in a convective cooling system that is not equipped with a fan. However, most systems require a fan for proper operation. Horizontal mounting is not recommended unless forced air cooling is provided at a rate of 30 cubic feet per minute passing over the surface of the board.

INSTALLING THE ZT 8802

The fastest way to begin using the ZT 8802 is with the addition of development software available from Ziatech. The STD ROM development system allows you to download application software developed on an IBM PC (or equivalent) through a serial port onto the ZT 8802. The personal computer is used as a development station to create, download, and debug applications written in assembly, C, and other high level languages.

Ziatech's DOS for the ZT 8802 is intended for designers who wish to develop an application using a high-level language or for systems requiring a resident operating system (to support disk subsystems, for example). Ziatech DOS is an MS-DOS operating system that resides on the ZT 8802 and is able to run most PC software.

In addition to high level language support, Ziatech DOS includes an extensive base of easily integrated software, such as support for the following:

- EGA and VGA graphics (ZT 8844 and ZT 8980/8981/8982)
- Floppy disks subsystems (ZT 8950 series)
- Fixed disk subsystems (ZT 8952/8953)
- SCSI-2 interface (ZT 8956)
- RAM and EPROM disks (ZT 8825 and on-board ZT 8802)
- Flash memory in-circuit programming (ZT 8825 and on-board ZT 8802)
- Centronics printer interface (ZT 88CT75)
- IEEE 488 (ZT 8847 and ZT 8848)
- Serial (ZT 8840, ZT 8841, ZT 8932 [quad serial], ZT 88CT75, zSBX 32, and on-board ZT 8802)
- Real-time clock (on-board ZT 8802)
- Digital signal processing (ZT 89CT30)
- Motion control (ZT 8931)
- Parallel processing (ZT 8832)
- PCMCIA (ZT 8921)
- GE FANUC interface (ZT 88CT93)

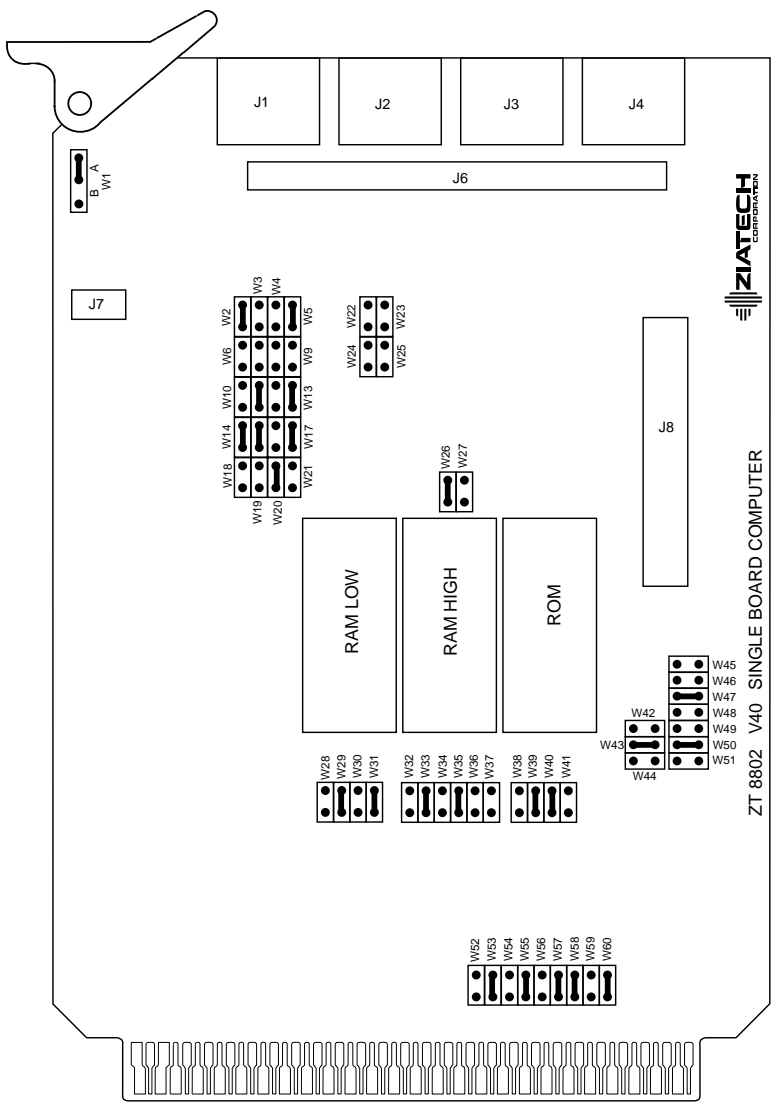


Figure 2-1. STD ROM (32K ROM/128K RAM) Configuration.

Configuring the ZT 8802 for STD ROM

The STD ROM development system is available as an option to the ZT 8802 for software development. If STD ROM is ordered along with the ZT 8802, the board is preconfigured and tested at the factory prior to shipment. In STD ROM configuration, all 48 lines of I/O are available for application use. If the ZT 8802 and STD ROM are ordered separately, or if the system has been altered or the ZT 8802 rejumped and the system does not function properly, refer to Figure 2-1 and check the following configuration requirements.

STD ROM Memory Requirements

- The STD ROM debug monitor is shipped in a 32 Kbyte EPROM. Install this EPROM into the 32-pin socket at location 5F, right justified, with the board oriented component side up, goldfingers to the left (see Figure 2-1).
- STD ROM requires 2 Kbytes of memory, from address 0h through 7FFh, for program use. The 128 Kbyte static RAM shipped with the STD ROM system should be sufficient for both debug and application program memory. Install this RAM into socket location 3F.
- If more application program RAM memory is required, a larger RAM may be installed in 3F, or an additional RAM may be installed in 4F. The ZT 8802 is designed to support four different memory configurations: one 128 Kbyte RAM; two 128 Kbyte RAMs (256 Kbytes); one 512 Kbyte RAM; or two 512 Kbyte RAMs (1 Mbyte). Smaller RAM sizes, such as 32 Kbytes, are redundantly mapped within each 128 Kbytes. PROM sizes are selectable between 128 and 512 Kbytes. Smaller sizes are redundantly mapped.

STD ROM Cable Requirements

- STD ROM requires a serial link between connector J7 and the IBM PC or compatible. The ZT 90069 cable shipped with the STD ROM system should be used for this purpose. Plug this cable into connector J7 of the ZT 8802. Plug the 25-pin D-shell connector into COM1 of a PC compatible.
- If your PC has a 9-pin serial cable (for example, IBM AT®), then an adapter cable is required for the 9-pin to 25-pin conversion.

STD ROM Jumper Configuration

The following jumper configuration should be used for STD ROM. Refer to Figure 2-1 on page 2-8 for an illustration of this jumper configuration. This configuration is for a 32K ROM (not Flash), 128K RAM system. Note that the 32K ROM is redundantly mapped four times within the top 128K of memory.

INSTALL: W1A, 2, 5, 11, 13-15, 17, 20, 26, 29, 31, 33, 35, 39, 40, 43, 47, 50, 53, 55, 57, 58, 60

REMOVE: W1B, 3, 4, 6-10, 12, 16, 18, 19, 21-25, 27, 28, 30, 32, 34, 36-38, 41, 42, 44-46, 48, 49, 51, 52, 54, 56, 59

STD ROM Operation

Please refer to your STD ROM documentation for software installation and operation procedures.

If the system is not working, check the following:

1. If a PC is used and it has more than one 25-pin male connector, be sure the serial cable is plugged into COM1.
2. Check to see the EPROM and RAM chips are installed in the proper sockets. EPROM should be installed in socket 5F. RAM should be installed in socket 3F.
3. Check pin 1 orientation of the installed EPROM and RAM(s). Pin 1 should be to the left, with the board oriented component side up, goldfingers to the left. It should be right-justified in the socket. This leaves four unused pins on the left side of the socket at 5F.
4. Verify that the jumpers are properly installed, particularly those associated with the socket and memory configurations, that is, W19-W21, W30-W31, W34-W35, W38-W44, W28-W29, W32-W33, and W22-W25.

Configuring the ZT 8802 for DOS

Ziatech's DOS is an optional MS-DOS operating system available for the ZT 8802 processor board. If the ZT 8802 and DOS are ordered together, the board is properly configured and tested as a system prior to shipment. If the ZT 8802 and DOS are ordered separately, or the ZT 8802 was altered in any way after shipment, you can find instructions to install and boot DOS on the ZT 8802 in the *Ziatech Industrial Computer System Manual*. Refer to Figure 2-2 on page 2-15 for an illustration of the correct DOS jumper configuration. Note that 40 lines of digital I/O are nominally available in a DOS system.

DOS Memory Requirements

The DOS/BIOS software is shipped in one 128 Kbyte Flash memory for installation onto the ZT 8802 at socket location 5F (see Figure 2-2 on page 2-15). Install the Flash only at a static-free workstation. Orient pin 1 properly, to the lower left with the board placed component side up, goldfingers to the left.

Ziatech DOS requires at least 128 Kbytes of static RAM, although the DOS system is shipped with 256 Kbytes. The two 128 Kbyte static RAMs are installed in socket locations 3F and 4F, occupying the memory address range from 0 through 3FFFFh. Again, be sure to orient pin 1 consistently with the Flash's pin 1.

DOS requires battery-backed RAM for system configuration variables; the top 32 Kbytes of a 128, 256, or 512 Kbyte system are designated as the R: drive and also retain system configuration information. For 1 Mbyte systems, the system configuration information is contained in the top 384 Kbytes of memory above the 640 Kbyte system RAM. In two-RAM (256 Kbyte/1 Mbyte) systems, socket 4F must be battery backed by installing W33 and removing W32. In single RAM (128/512 Kbyte) systems, socket 3F must be battery backed by installing W29 and removing W28. A 1 Amp-hour lithium battery is shipped with the DOS system to back up the system information and optionally, all RAM on board. It should be securely placed into the battery socket and held fast with double-sided tape underneath.

Note that when W33 is installed, the whole static RAM is battery backed in socket 4F. W29 will battery back all of the RAM in 3F.

DOS Cable Requirements

The RS-232-C serial interface signals for COM1 and COM2 are available at frontplane connectors J4 and J3, respectively. The ZT 90136 serial cable can be used with these ports. It connects to the 10-pin dual-row headers of J3 and J4 and has a 9-pin female D-shell connector on the other end.

If the DOS system is a PC-assisted (PCA) system, a serial link is also required between the ZT 8802 and a terminal or host PC. Refer to the *Ziatech Industrial Computer System Manual* for cable requirements and software information. The ZT 90069 cable is used to connect J7 to the COM port of the host system. If your PC has a 9-pin serial cable (for example, IBM AT), then an adapter cable is required for the 9-pin to 25-pin conversion.

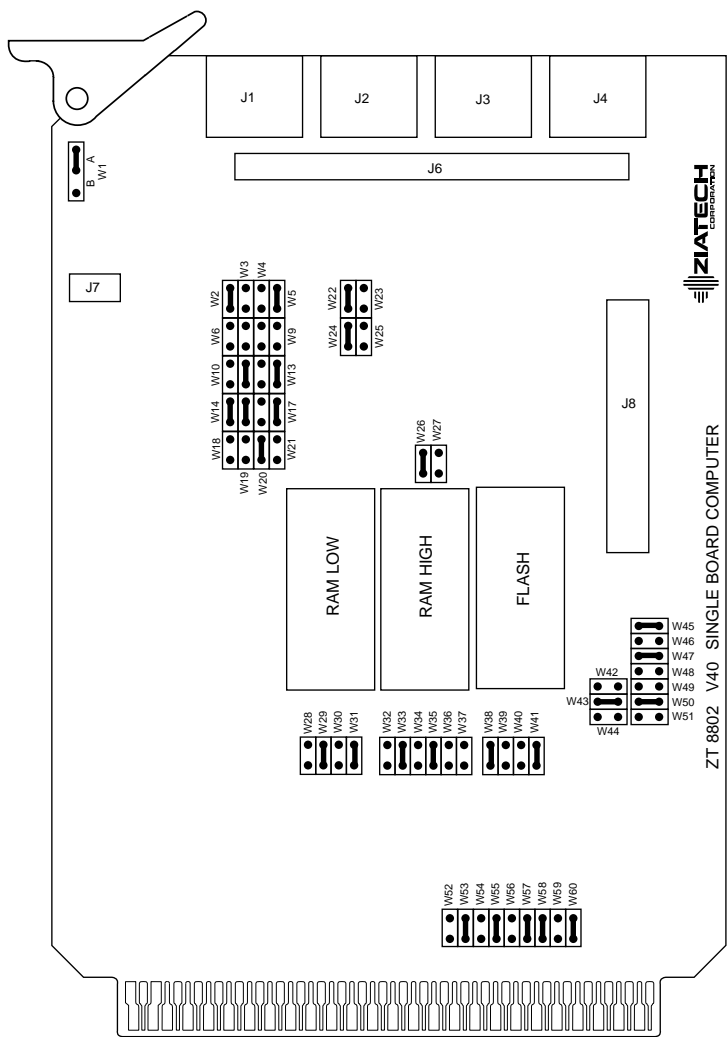
DOS Jumper Configuration

The ZT 8802 as shipped from Ziatech should be properly configured for a DOS system if the board was ordered with the DOS option. The following is a list of the jumpers assigned at the factory prior to shipment. Refer to Figure 2-2 on page 2-15 for an illustration of this jumper configuration. This configuration is for a 128K Flash, 2x128K (256K) RAM system. Consult Figure 2-3 on page 2-16 for a 512K ROM/1 Mbyte RAM system configuration.

INSTALL: W1A, 2, 5, 11, 13-15, 17, 20, 22, 24, 26, 29, 31, 33, 35, 38, 41†, 43†, 45, 47, 50, 53, 55, 57, 58, 60

REMOVE: W1B, 3, 4, 6-10, 12, 16, 18, 19, 21, 23, 25, 27, 28, 30, 32, 34, 36, 37, 39, 40†, 42, 44†, 46, 48, 49, 51, 52, 54, 56, 59

† Install W41 and W44 and remove W40 and W43 for Flash in-circuit program capability. The +12 V power supply must be regulated within 5% (11.4 V to 12.6 V) for proper Flash memory operation.



Note: INSTALL W44, W41; REMOVE W43, W40 for Flash in-circuit program capability.

Figure 2-2. DOS (128K Flash/256K RAM) Default.

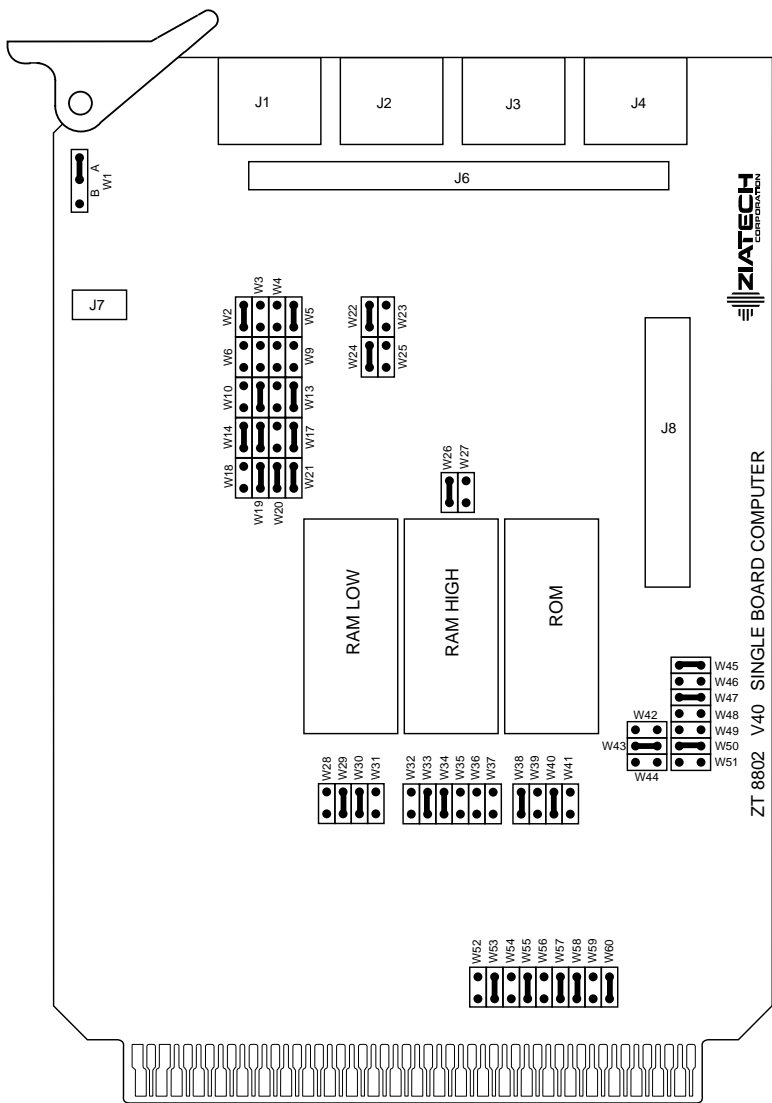


Figure 2-3. DOS (512K ROM/1M RAM) Configuration.

Powering Up The DOS System

Be sure the ZT 8802 is seated securely in the card cage and the power switch is off. Plug the card cage into a 120 VAC source. Refer to the following instructions to find those that are appropriate to your configuration (PC-Assisted with a host computer, PC-Assisted with a video board ["Stand-Alone Operation"], or Automation Engine).

PC-Assisted with a host computer - An IBM PC or compatible is used to communicate with the ZT 8802 DOS system.

1. Connect the DOS system's serial cable from the proper connector (J7) on the ZT 8802 to COM1 on the IBM or compatible PC with the ZT 90069 cable.
2. Install the Host Development Software diskette into drive A of your IBM or compatible PC.
3. Type A:VSC and press return. The screen should indicate VSC is installed.
4. Press the ALT + SPACE keys to switch to the DOS system screen. The screen should be clear on the DOS system side.
5. Power on the DOS system.
6. The system configuration appears, followed by the "ZT P:>" prompt.

If you need further assistance, refer to your Ziatech system manual.

Stand-Alone Operation - If the ZT 8802 was purchased with a video board, the system may be operated as a stand-alone system with complete keyboard and video support.

1. Configure the video board for the type of monitor desired:
 - a) The ZT 8844 EGA video board is shipped configured for a monochrome monitor. If your monitor is not monochrome, refer to the *ZT 8844 Hardware Manual* for alternate jumper configurations.
 - b) The ZT 8980 and ZT 8981 video boards are shipped configured for VGA and EGA color monitors, respectively. Refer to your hardware manual for specific jumper configurations.
 - c) Be sure the video board is installed into the card cage along with the ZT 8802 and the cables to the display and keyboard are attached.
2. Power on the system. You should see the system configuration list and a "ZT P:>" prompt on your display. Operation is then identical to a PC-compatible type computer.

The Automation Engine is available for OEM system designers or high volume users of the ZT 8802 DOS system. The ZT 8802 is shipped with a license for MS-DOS and it is used for systems with completed application software. It is assumed here that the user is familiar with the ZT 8802 DOS system.

MEMORY ADDRESSING

Figures 2-4 through 2-11 show the memory addresses occupied by the ZT 8802 for DOS and STD ROM systems.

Access to on-board memory and the backplane is through the full 20-bit memory address, allowing for 1 Mbyte of memory in the system. On-board memory consists of three 32-pin byte-wide sockets. One of these sockets accepts from 32 to 512 Kbytes EPROM/Flash and the other two sockets accept either 128 or 512 Kbyte RAMs. All RAM may be battery backed. The EPROM space is selectable between 128 and 512 Kbytes.

Memory access times require the use of 200 ns parts or faster for both RAM and ROM accesses.

The ZT 8802 has four memory pages that are either software selectable or hardwired. This allows up to 1 Mbyte and 512 Kbyte of EPROM/Flash to be mapped within the 1 Mbyte address space of the V40. Each of these pages is referred to as a *MODE*. The memory map has also been optimized for one of four RAM memory configurations: 128 Kbytes (one 128 Kbyte part), 256 Kbytes (two 128 Kbyte parts), 512 Kbytes (one 512 Kbyte part), and 1 Mbyte (two 512 Kbyte parts). It is possible to use smaller parts (such as 32 Kbyte RAMs) when configuring the board for 128 Kbyte parts as a method to reduce costs. Using smaller parts causes the part to be redundantly mapped within the 128 Kbyte memory space. If you are using DOS, do not use RAMs smaller than 128 Kbytes.

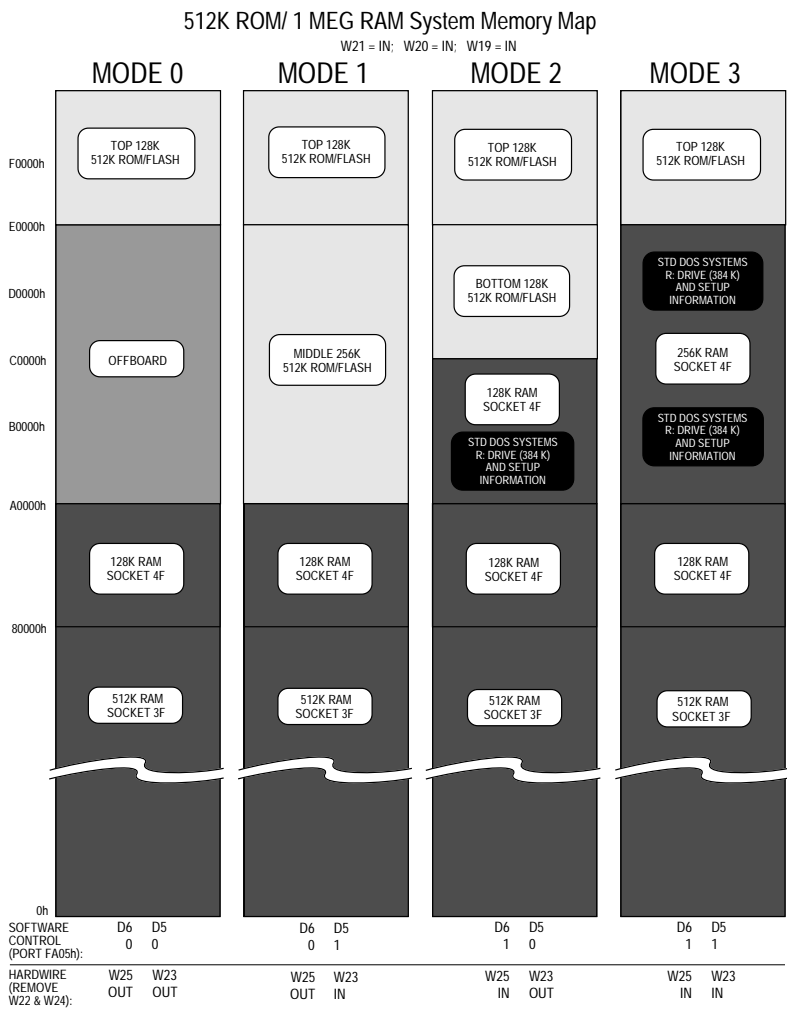


Figure 2-4. 512K ROM/1M RAM System Memory Map.

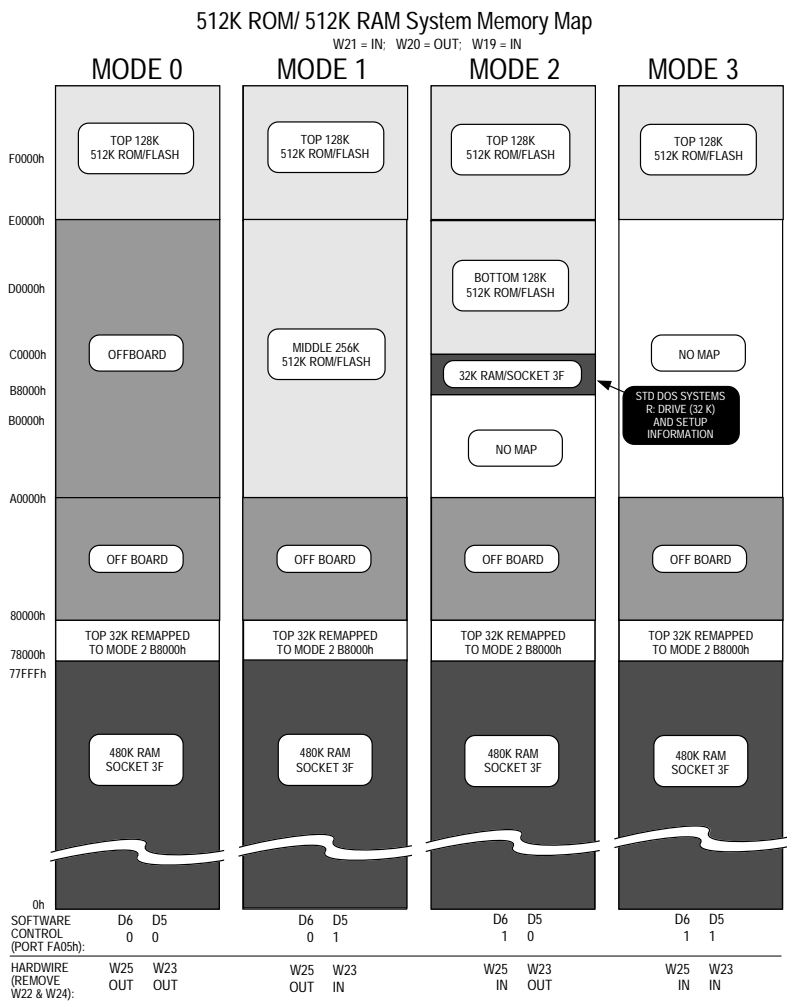


Figure 2–5. 512K ROM/512K RAM System Memory Map.

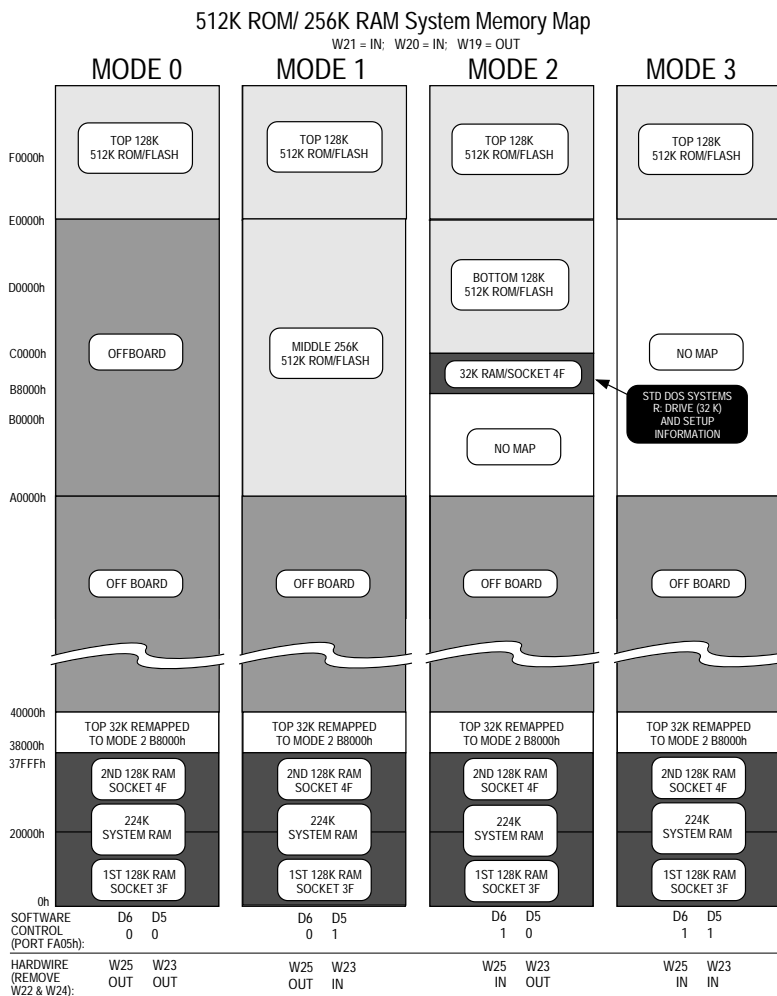


Figure 2–6. 512K ROM/256K RAM System Memory Map.

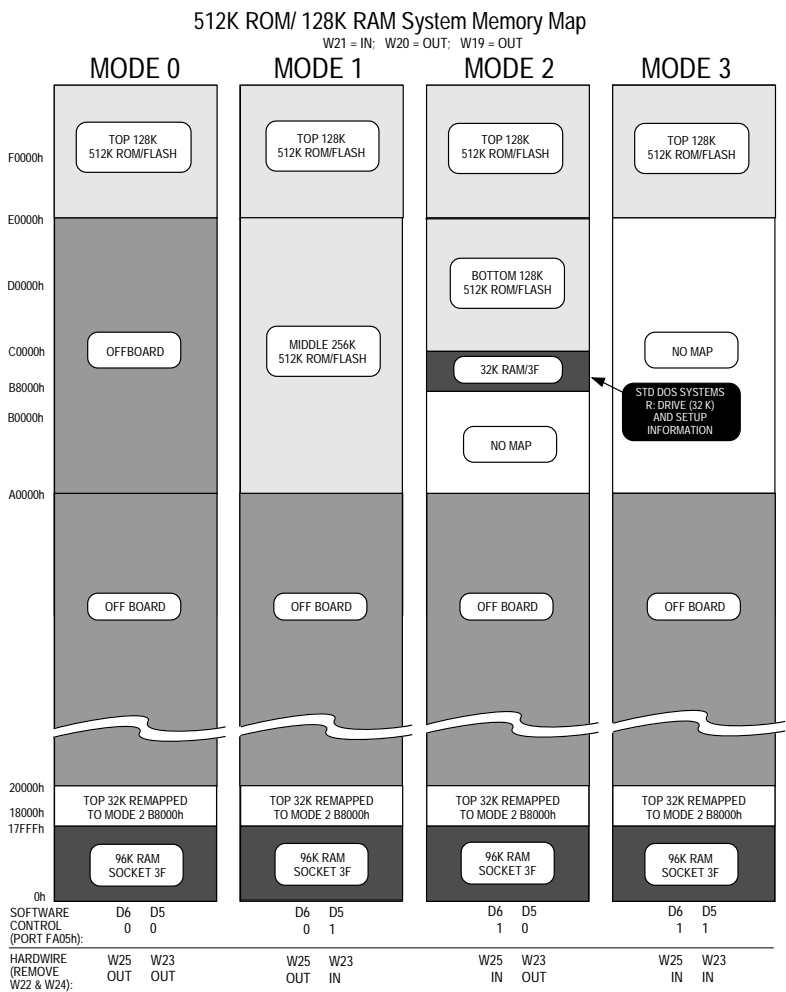


Figure 2–7. 512K ROM/128K RAM System Memory Map.

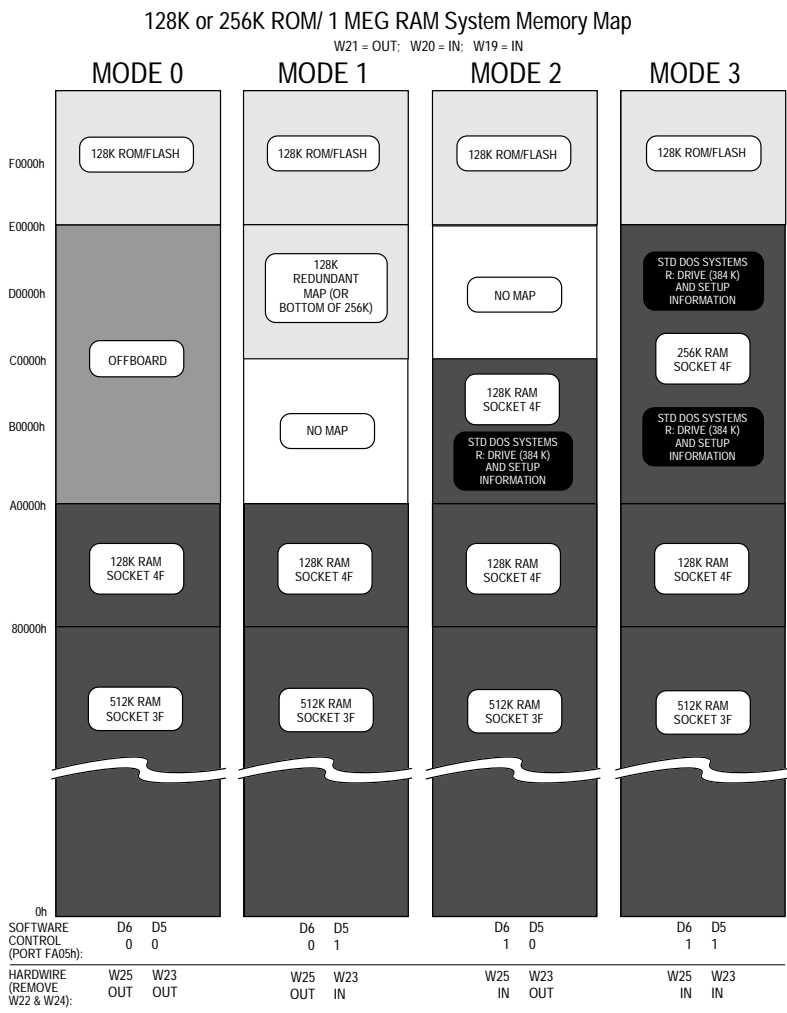


Figure 2-8. 128K/256K ROM-1M RAM Memory Map.

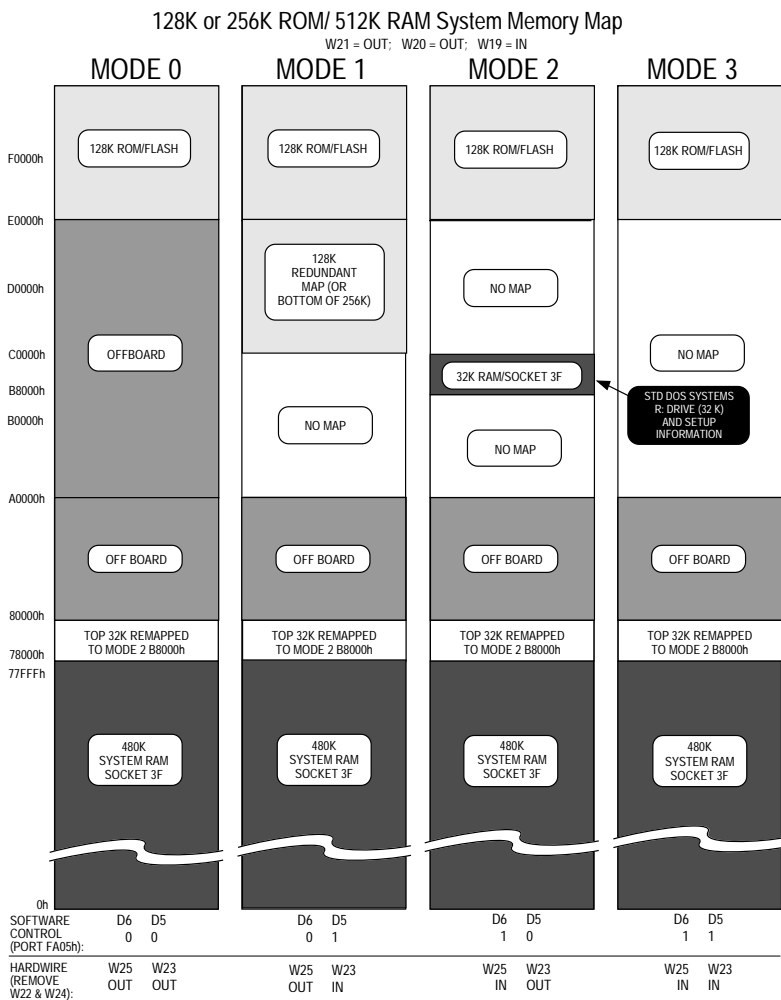


Figure 2–9. 128K/256K ROM-512K RAM Memory Map.

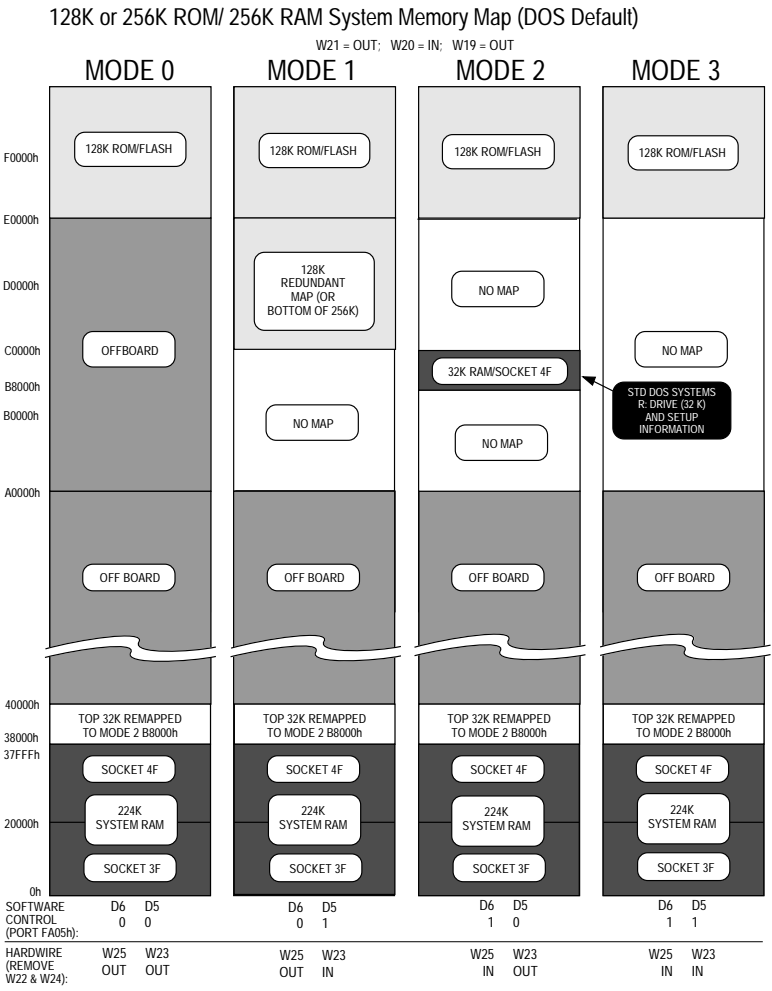


Figure 2–10. 128K/256K ROM-256K RAM Memory Map.

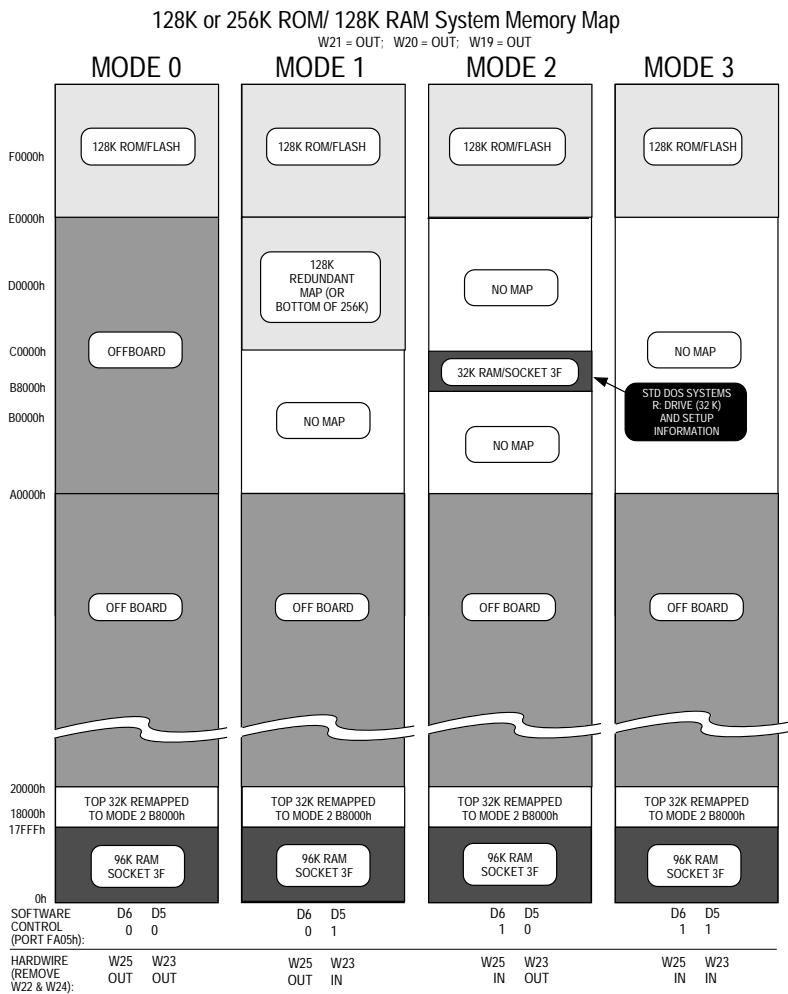


Figure 2–11. 128K/256K ROM-128K RAM Memory Map.

Memory Modes

The four memory modes (MODE 0 - MODE 3) allow the ZT 8802 to bank switch extra RAM and EPROM/Flash into the 256 Kbyte memory space between A0000h and DFFFFh. Under software control, two parallel port bits are used to select the MODE in which the memory is configured. Bits 5 and 6 of I/O address 0FA05h are used to select the mode. 00b written selects MODE 0; 01b selects MODE 1; 10b selects MODE 2; and 11b selects MODE 3. Other bits are used within this register, which requires that software read the port before writing a new mode to preserve the other bits. We also recommend you disable interrupts (CLI) before modifying this register. See the "Control Port" section, page 3-31, for further details.

In all modes, the top 128 Kbytes of EPROM/Flash are always accessible, as is any system or application memory below A0000h.

MODE 0

In MODE 0, memory accesses to A0000h-DFFFFh go off-board. Note that in DOS systems this memory address range is where video boards and other memory mapped devices are located. MODE 0 is the default mode at power up.

MODE 1

MODE 1 is used when a 512 Kbyte EPROM is installed to reach the middle 256 Kbytes of the EPROM at addresses A0000h-DFFFFh. In this mode, a total of 384 Kbytes of EPROM space exists from A0000h through FFFFFh. Below A0000h, system RAM is available up to the amount of RAM installed.

MODE 2

MODE 2 is used when a 512 Kbyte EPROM/Flash is installed to access the bottom 128 Kbytes of the part, which is accessible between C0000h through DFFFFh. In MODE 2, the top 128 Kbytes of the second 512 Kbyte RAM in a 1 Mbyte system are accessible at addresses A0000h through BFFFFh. Note that in MODE 2 the top 128 Kbytes of the 512 Kbyte EPROM are still accessible between E0000h through FFFFFh.

MODE 3

MODE 3 is used when two 512 Kbyte RAMs are installed in order to access the middle 256 Kbytes of the second RAM. This memory is accessible between A0000h through DFFFFh. MODE 2 is used to access the top 128 Kbytes of the second RAM.

The memory map figures beginning on page 2-20 show the memory modes used for the four memory configurations.

Mode Hardwiring

Alternatively, the ZT 8802 can be hardwired into one of these modes via jumpers W22-W25. This frees these two parallel port bits for other uses. To hardwire the board into a particular mode, remove W22 and W24. W23 and W25 are then used to select the mode. Removing W23 and W25 forces the board in MODE 0. Installing only W23 forces the board in MODE 1. Installing only W25 forces the board in MODE 2. Installing both W23 and W25 forces the board to always be in MODE 3. DOS users must allow software control over the mode bits and cannot hardwire to a particular mode. The default for STD ROM users is to remove W22-25, forcing the board to always be in MODE 0.

I/O ADDRESSING

Figure 2-12 on page 2-31 shows the I/O addresses occupied by the ZT 8802 for DOS and STD ROM systems.

I/O accesses are made via the full 16-bit I/O address, allowing for 64 Kbytes of I/O addresses. Eight-bit addressed I/O boards are also compatible with the ZT 8802, provided they decode the IOEXP signal on the backplane low when generating board select. IOEXP is driven low only when I/O addresses FC00h through FFFFh are accessed. Eight-bit addressed boards that decode IOEXP low are then accessed only within the FC00h-FFFFh range and are not redundantly mapped throughout the I/O map. For example, an 8-bit board that decodes an address of 80h, as well as decodes IOEXP low, is accessible from the ZT 8802 at address FC80h, but not at 80h, which keeps the I/O map below FC00h free for other 16-bit decoders.

JUMPER OPTIONS

The ZT 8802 includes many jumper options to tailor the operation of the board to the requirements of specific applications. Refer to Appendix A for a full description of configurable jumpers.

FFF0-FFFFh	V40 Configuration	IOEXP driven low
FC00-FFEFh	STD Bus	
FB80-FBFFh	SBX 1 Chip Select	
FB00-FB7Fh	SBX 0 Chip Select	
FA80-FAFFh	RTC	
FA00-FA7Fh	Parallel ASIC	
0400-F9FEh	STD Bus	
03F8-03FFh	COM 1	
0300-03F7h	STD Bus	
02F8-02FFh	COM 2	
00E0-02F7h	STD Bus	
00D0-00DFh	DMA Controller	software mapped
00C0-00CFh	STD Bus	
00B0-00BFh	V40 Serial Port	software mapped
0050-00AFh	STD Bus	
0040-004Fh	Counter/Timers	software mapped
0030-003Fh	STD Bus	
0020-002Fh	Interrupt Controller	software mapped
0000-001Fh	STD Bus	

Figure 2-12. I/O Map.

THEORY OF OPERATION

Contents	Page
OVERVIEW	3-3
COMMONLY ASKED QUESTIONS	3-4
RELATIVE MICROPROCESSOR PERFORMANCE	3-8
STD BUS COMPATIBILITY	3-9
STD INTERFACE	3-10
STD Bus Memory Transfers	3-10
STD Bus I/O Transfers	3-10
IOEXP	3-11
MEMEX (BHE*)	3-11
DIRECT MEMORY ACCESS (DMA)	3-12
Advantages of DMA	3-12
DMA Operation	3-13
I/O POINTS	3-15
Input/Output Operation	3-16
Outputting Data	3-16
Outputting Data With Readback	3-16
Inputting Data	3-17
Masking Operation	3-18
Event Sense Operation	3-19
Electrical Specifications	3-21
Reset Operation	3-21
Power Supply Lines	3-22
Connectors	3-22
SERIAL COMMUNICATIONS	3-23
INTERRUPTS	3-24
Interrupt Request Assignments	3-24
Polled Interrupts on the STD Bus	3-26
STD Bus Vectored Interrupts	3-28

STD Bus Cascaded Interrupts	3-29
Non-Maskable Interrupts	3-30
RESET	3-31
CONTROL PORT	3-31
Hardwiring Features vs. Software Control	3-33
POWER-FAIL PROTECTION	3-34
DC Power Fail	3-34
BATTERY	3-35
STATUS INDICATOR (LED)	3-37
CMOS VERSION OF THE ZT 8802	3-37
Added Features	3-37

OVERVIEW

This chapter begins by answering commonly asked questions about the ZT 8802 and ZT 88CT02. It then describes the following system level issues:

- Processor performance compared to the IBM PC XT®
- STD bus compatibility
- STD bus memory and I/O transfers
- Direct Memory Access (DMA) support and benefits
- I/O operation, including masking and event sense
- Serial communications using RS-232-C
- Expanding the ZT 8802 interrupt structure
- Processor reset
- Hardwiring vs. software control through the control port
- Methods of battery backup with DC power-fail detection
- Battery life
- Status indicator access (LED)
- CMOS version available for extended temperature and low power

COMMONLY ASKED QUESTIONS

1. **What are the differences between the ZT 8802 and ZT 8901 STD 32 processor boards?**

The ZT 8901 is designed with the NEC V53 processor running at 16 MHz. The ZT 8802 is designed with an 8 MHz V40 processor. The ZT 8901 can perform 16-bit transfers on the STD 32 bus, while the ZT 8802 has only 8-bit transfer capability. Both boards have 48 points of Opto 22 compatible I/O. The connector for this I/O is in the same physical location on both boards to allow for modular system integration.

2. **What are the differences between the ZT 8802 and ZT 8801?**

The ZT 8802 is a superset of the ZT 8801, adding two DOS compatible COM ports for serial interfacing. The ZT 8802 does not have RS-485 support for the V40 serial port. The ZT 8802 uses surface mount technology, but the mechanical positions for the digital I/O and SBX connectors are identical.

3. **Is the ZT 8802 compatible with the mechanical specification of the STD bus?**

Yes. The ZT 8802 uses no extensions or porches to increase the board size.

4. Is the V40 pin-compatible with the 80188?

The V40 and 80188 are not pin-compatible. This means an 80188 cannot be plugged into the V40 socket. This is unlike the V20 and V30, which are interchangeable with the 8088 and 8086, respectively.

5. What are the hardware differences between the V40 and the 80188?

One of the most notable differences is that the V40 is fabricated with a CMOS process. CMOS technology provides an increase in both temperature range and noise immunity with a reduction in power consumption. The CMOS V40 has a maximum power dissipation of less than 1/2 W and a standby dissipation of less than 1/10 W. The 80188 has a maximum power dissipation of 3 W; that is, too hot to touch.

Other differences are as follows: the V40 includes an asynchronous serial port; the V40 interrupt controller and counter/timers have the same architecture as the interrupt controller and counter/timers used in the IBM Personal Computer; and the data transfer rate for the V40 DMA controller is twice as fast.

6. Is the V40 software compatible with the 8088 and 80188 microprocessors?

Yes. The V40 instruction set is 100% object code compatible with the 80188 instruction set. This means a program written for the 8088 or 80188 will execute on the V40. Application software using the peripherals internal to the 80188 requires some modification.

7. What are the V40 instructions not in the 8088 or 80188 and how are they used?

The V40 instruction set is a superset of the 8088 and 80188. This means the V40 includes all of the instructions found in these microprocessors plus a few more. The added instructions are outlined below.

The following instructions are useful in testing and setting status bits for I/O operations and in bit manipulation for graphics applications.

INS	Insert bit field
EXT	Extract bit field
TEST1	Test bit
SET1	Set bit
CLR1	Clear bit
NOT1	Complement bit

The instructions shown below are useful for manipulating binary numbers in a decimal format.

ADD4S	BCD string addition
SUB4S	BCD string subtraction
CMP4S	BCD string comparison
ROL4	Rotate BCD digit left
ROR4	Rotate BCD digit right

The string I/O instructions shown below can be combined with the repeat prefixes for high speed data transfers between I/O and memory.

INM	String input
OUTM	String output

Other instructions not found in the 80188 instruction set are listed below.

REPC	Repeat while carry set
REPNC	Repeat while carry cleared
FPO2	Floating point operation 2
BRKEM	Break for emulation mode
RETEM	Return from emulation mode

RELATIVE MICROPROCESSOR PERFORMANCE

The "Norton Utilities System Information Version 6.0" was used to measure the ZT 8802 processor performance relative to that of the IBM PC. The test compares several processing tasks; test results are presented in Table 3-1. These benchmarks assume one (1) I/O wait state and zero (0) on-board memory wait states.

Table 3-1
Processor Speed Comparison.

Test	ZT 8802 Average Test Score Relative to 4.77 MHz PC/XT
Processor Speed Benchmark	1.9
Disk Speed (ZT 8952)	4.5
Overall Performance Index	2.7

STD BUS COMPATIBILITY

The ZT 8802 is fully compatible with Revision 2.3 of the *STD-80 Series Bus Specification and Designer's Guide* (Ziatech document number ZT MSTD80). This revision of the Bus Specification includes definition of two new backplane interrupt request signals, INTRQ1* and INTRQ2*, which replace the signals RESERVED and CNTRL*, respectively. Prior to Revision 2.3, only one backplane interrupt request INTRQ* existed, requiring frontplane cabling or sharing of the interrupt if more than one backplane interrupt was in the system. DOS uses INTRQ1* for the keyboard interrupt request from the ZT 8980 Super Video Graphics Adapter (VGA) card. See the discussion of the interrupt system beginning on page 3-24 for details on these interrupts.

STD INTERFACE

The ZT 8802 performs 8-bit I/O and memory transfers to the STD bus whenever an off-board address is decoded for either cycle type. The ZT 8802 performs a read or write transfer in four STD bus clock cycles, in accordance with the *STD-80 Series Bus Specification and Designer's Guide*, Revision 2.3. STD bus peripherals that insert wait states are accommodated by programming the V40 for one internal wait state. This allows the off-board peripheral to generate additional wait states should they be needed.

STD Bus Memory Transfers

The ZT 8802 accesses off-board memory whenever the address generated is not decoded as an on-board memory location. The V40 drives only 20 bits of address on memory transfers; the ZT 8802 drives an additional four address lines (A20-A23) for a total of 24 bits of address. A20-A23 are driven to logical 0 for off-board accesses.

STD Bus I/O Transfers

The ZT 8802 accesses off-board I/O whenever the address generated is not decoded as an on-board I/O location. The ZT 8802 supplies a full 16-bit I/O address for off-board transfers.

IOEXP

The *STD-80 Bus Specification and Designer's Guide* never clearly defined IOEXP usage. The *STD 32 Bus Specification* defines that IOEXP be driven low in the I/O address space of FC00h-FFFFh by the current bus master. With this mechanism, boards that decode only 8 bits or 10 bits of address and that can use IOEXP in their board select logic can be logically mapped in this range (FC00h-FFFFh). IOEXP is used as an extra address bit in this case and keeps the 8- or 10-bit address board from being redundantly mapped in the lower I/O space.

Even though the ZT 8802 is not an STD 32 processor, this definition for IOEXP is used to ease system integration.

MEMEX (BHE*)

The MEMEX signal is passively pulled high by the ZT 8802 or it may be optionally tied to ground via jumper W52. STD 32 peripheral boards interpret this signal as Byte High Enable (BHE*) during 16-bit transfers. When this signal is low, STD 32 peripherals that dynamically support 16-bit transfers use this signal to gate the upper data bits of the transfer onto D8-D15. For this reason, we recommend that W52 not be installed when interfacing to STD 32 peripherals. In this mode, BHE* is always high.

DIRECT MEMORY ACCESS (DMA)

The ZT 8802 supports Direct Memory Access (DMA) transfers between local memory and STD bus system memory or I/O under the supervision of an STD bus DMA controller, such as the ZT 8950. The following discussion covers the advantages and operation of an STD bus DMA controller with respect to the CPU.

Advantages of DMA

The use of DMA can greatly increase performance in a system where block transfers of memory or I/O data are often performed. For example, if a disk subsystem is present on the STD bus, large data transfers from the disk controller to system memory and vice versa frequently take place. The use of DMA can allow these transfers without requiring CPU time, thus improving system throughput.

The CPU normally performs data transfer. In a memory to I/O transfer, the CPU must first fetch the memory read instruction from program memory, read the data from memory, temporarily store the data, fetch the I/O instruction from program memory, and then write the data to I/O. This process becomes very time-consuming when moving large blocks of data, as is required for disk I/O.

The DMA controller speeds up this operation because the controller contains source and destination address counters that eliminate the need to fetch program instructions; data transfers are performed sequentially under the supervision of the controller. A single cycle transfer enables the data from the source onto the bus and to the destination without temporarily storing it.

DMA Operation

Figure 3-1 shows the interface between the ZT 8802 and an STD bus DMA controller, such as the ZT 8950. The signals shown are required for proper operation of devices on the STD bus during DMA cycles.

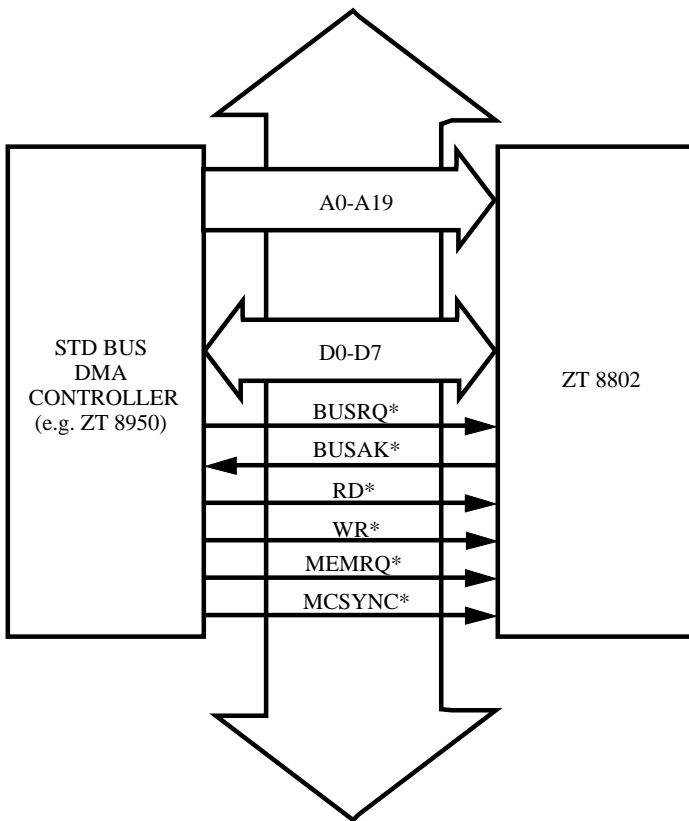


Figure 3-1. DMA With STD Bus Controller.

The DMA cycle is initiated when the controller asserts the bus request signal BUSRQ* on the STD bus. The ZT 8802 responds to this request by waiting for the CPU to finish the current instruction, then acknowledges the release of the bus by asserting the bus acknowledge signal BUSAK*. Simultaneously, the ZT 8802 turns the address buffers inward to allow access to on-board memory by the DMA controller. The controller is then free to transfer data between an STD bus board and the ZT 8802 or between two STD bus boards. Note that in the case of the ZT 8950, the I/O device may be a separate board. The ZT 8950 has two frontplane connectors for external DMA devices in addition to the on-board (ZT 8950) floppy disk controller.

The STD bus DMA controller must meet the timings for read and write cycles as defined by the *STD-80 Series Bus Specification and Designer's Guide*.

I/O POINTS

The 48 input/output points on the ZT 8802 are implemented in one Ziatech 16C49 Parallel Interface Adapter (PIA). The PIA has 48 I/O points in six groups (ports) of eight I/O points each. Each I/O line is illustrated in Figure 3-2, "Typical I/O Circuit."

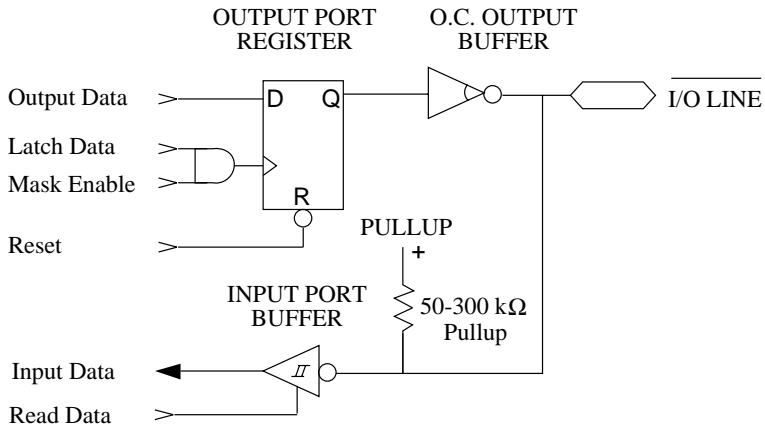


Figure 3-2. Typical I/O Circuit.

Each typical I/O circuit consists of an output register, open collector output buffer, pullup, and input buffer. Input and output operation is described on the following pages. Applications requiring pullups at the I/O interface should provide appropriate external termination. The internal pullup provided by the 16C49 should not be used for this purpose.

Input/Output Operation

Each 16C49 I/O circuit is capable of outputting data, outputting data with readback, and inputting data, as described below.

Outputting Data

Outputting data is accomplished by writing the output data to a given port, thereby causing the latch data signal to capture the output data into the output port register. The register output is buffered by an inverting open collector output buffer before driving the output signal. An integral pullup resistor ensures that a valid high can be measured when the output is not sinking current while in the de-asserted or "off" state.

Note: Each output port has a mask enable bit that can prevent inadvertent writes. The mask enable bits are controlled from the mask port and are unmasked to allow writes after power up or reset.

Outputting Data With Readback

Outputting data with readback can be accomplished by outputting data as described above and then reading the input data, thereby causing the read data signal to enable the input port buffer.

Inputting Data

Inputting data is accomplished by reading the input data from a given port, thereby causing the read data signal to input data from the inverting input port buffer. An integral pullup resistor ensures that a valid input is read if the input signal is not connected.

When inputting data, the associated circuit must not be used as an output. This allows the output buffer to be inactive, thereby not contending with the input signal. The output circuits are inactive when a logical 0 is output, after reset or power up.

When using a port configured with some output and some input circuits, care must be taken to ensure that any circuits used as inputs are always written with a 0.

Masking Operation

A separate mask port is provided to prevent inadvertent writes to the individual I/O ports. Not required for normal use, this optional port can help ensure system integrity if the system software were to accidentally do I/O port writes. Power up or reset leaves the mask port enabled for I/O writes.

Table 3-2 below illustrates the data bit and mask port relationships.

Table 3-2
Mask Port.

Port	D7	D6	D5	D4	D3	D2	D1	D0
5A07h (write) 5A07h (read)	E7-4 Int	E3-0 0	Port 5 Port 5	Port 4 Port 4	Port 3 Port 3	Port 2 Port 2	Port 1 Port 1	Port 0 Port 0

The upper two bits in the mask register are used in conjunction with the event sense port. When writing, the upper two bits of the mask register select the polarity sensed by event sense inputs E0-E7, which are connected to Port 0 (J6 pins 1-8 for E0-E7, respectively). Bits 7 and 6 determine E7-4 and E3-0, respectively. Writing a 0 (power-up default) senses negative events (edges), while writing a 1 senses positive events.

When reading the mask register, the most significant bit (D7) returns the interrupt signal status on the interrupt output pin of the 16C49. A logical 1 means interrupt is asserted.

Event Sense Operation

Eight event sense inputs on the 16C49 PIA are connected in parallel to I/O points $\overline{I/O} \ 0-7$. The event sense circuit is illustrated below.

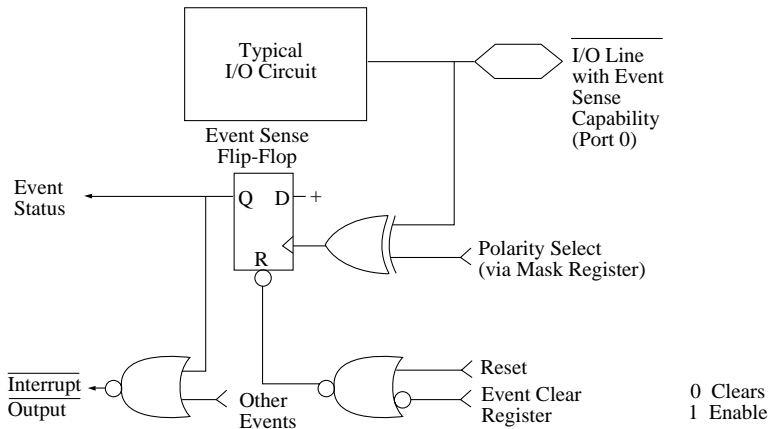


Figure 3-3. Typical Event Sense Circuit.

The event clear register has eight data bits corresponding to each of the eight event sense flip-flops. To enable operation, set the appropriate bit to a logical 1 (enable). Depending on the polarity selected via the mask register, a positive or negative transition on the I/O point clocks the event sense flip-flop. The event sense flip-flop status can be read via the event status register.

All eight event sense flip-flop outputs are ORed together to generate the hardware interrupt signal. Refer to Chapter 13 for additional information on programming for event sense operation.

Table 3-3 illustrates the data bit and event sense relationships.

Table 3-3
Event Sense Port.

Port	D7	D6	D5	D4	D3	D2	D1	D0
FA06h	E7	E6	E5	E4	E3	E2	E1	E0

When reading the event sense port, each bit set to a logical 1 indicates an event has occurred on that input.

When writing the event sense port, each data bit written with a logical 0 clears its corresponding event sense flip-flop. Each data bit of the event sense port must be written with a 1 to re-enable the corresponding event sense input after it is cleared or after power up or reset.

Electrical Specifications

The I/O line electrical specifications are listed in Table 3-4.

Table 3-4

I/O Line Electrical Specifications.

Parameter	Specification
Output Sink Current (Iol)	12 mA min.
Output Low Voltage (Vol at Iol)	.4 V max.
Internal Pullup	50 k Ω - 300 k Ω

Reset Operation

Each I/O circuit on the PIA is automatically reset by the reset circuit on the ZT 8802. The precision nonglitching reset circuit on the ZT 8802 prevents the output circuits from glitching on power up or power down. The reset circuit is active when the supply voltage is within 0-4.75 V.

After reset, the mask register is initialized to enable (unmasked) writing to the I/O registers and the event sense status register is cleared. All outputs are disabled (high) until written.

Power Supply Lines

When interfacing to an Opto 22 or equivalent I/O rack, the I/O rack requires +5 V to operate. The I/O rack typically has a two-position terminal block to which power can be supplied, or power can be supplied through the cable connected to the ZT 8802, via the optional ZT 2225 Industrial I/O Cable Adapter. Refer to page 13-16 for details on interfacing to I/O racks with the ZT 2225 and associated cables.

The +5 V power is available on pins 54-56 of J6. This power is protected by a 1 A miniature fuse. The fuse location on the ZT 8802 is designated F1. Replacement fuses can be purchased from Littelfuse, part #255-001.

Connectors

The ZT 8802 has one header designated J6 for 48 I/O points. Each I/O line on the ZT 8802 is connected to this ribbon cable compatible header. +5 V and ground are available at this interface for powering Opto 22 interfaces. The ZT 2225 Industrial I/O Cable Adapter can be used to make a transition from this 56-pin header to an Opto 22 compatible interface. Refer to page 13-16 for additional details and to page B-13 for J6 pin assignments.

Ziatech also offers the ZT 2226 24-Channel I/O Mounting Rack, which can be connected directly to the ZT 8802. It holds up to 24 I/O modules and can be panel mounted using its integral mounting hardware. A second ZT 2226 or other 24-channel I/O module mounting rack can be daisy-chained to the first ZT 2226 to provide a total of 48 I/O module positions.

In situations where it is impossible or undesirable to use the ZT 2226, Ziatech recommends the ZT 2223 Industrial I/O Adapter Board. The ZT 2223 provides many of the same features found on the ZT 2225, but allows the ZT 8802 to be cabled directly to non-Ziatech racks.

Combining a ZT 8802 with either I/O device can reduce the size of a control system and free up valuable enclosure space. See the "I/O Module Mounting Racks" section on page 13-14 for details.

SERIAL COMMUNICATIONS

The ZT 8802 has three asynchronous RS-232-C serial ports. The V40 serial port (available through connector J7) supports Transmit Data (TxD) and Receive Data (RxD) only. This port is not IBM PC (COM) compatible, but is supported by Ziatech's Device Driver Package (DDP), VSC under DOS, or VTI under STD ROM. Two DOS COM compatible serial channels are also provided via the 16C452. These channels are configured as RS-232-C and interface to 10-pin front-plane connectors J3 and J4. COM1 is available on J4, and COM2 on J3.

The ZT 90069 cable assembly may be used to interface between J7 and an external serial channel. This cable assembly interfaces the 3-pin serial interface to a standard 25-pin D-shell style connector.

INTERRUPTS

The ZT 8802 supports both maskable and non-maskable interrupts. This section discusses system level issues related to these interrupts. Refer to Chapter 7 for more information on the operation and programming of the maskable interrupt controller.

Interrupt Request Assignments

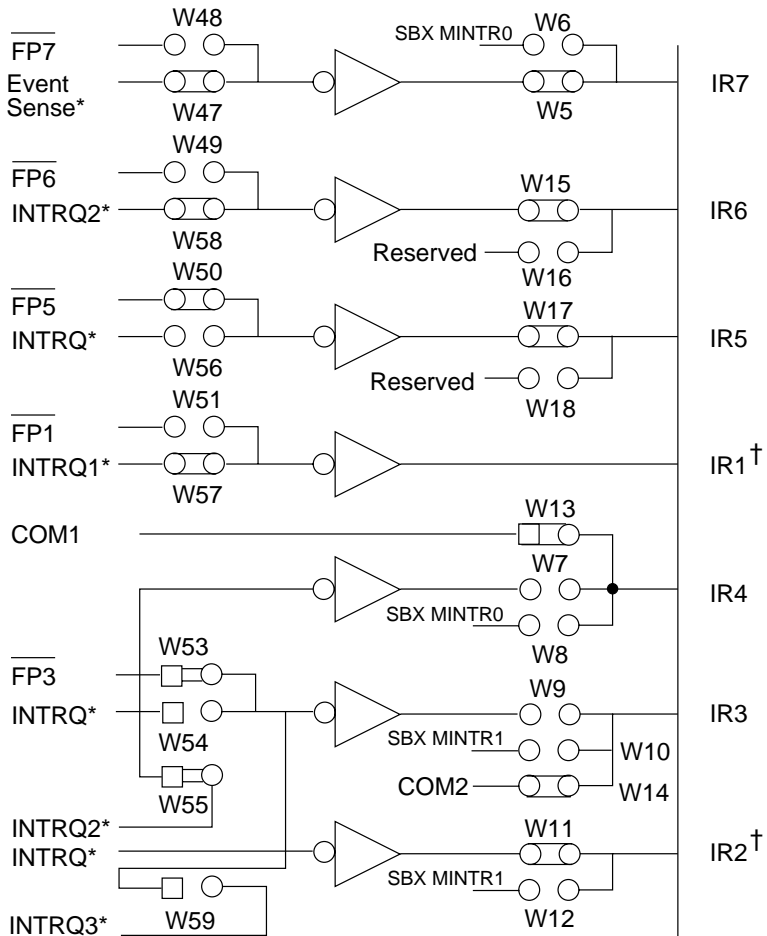
The 8259A Programmable Interrupt Controller (PIC) on board the ZT 8802 has seven interrupt input requests plus one internal V40 interrupt (TIMER0), each with a number of possibilities for an interrupt source. Figure 3-4 shows these possibilities, which are assigned via jumper selections W5-W18, W47-W51, and W53-W59. This figure indicates the factory default assignments.

Three of these interrupt assignments are critical for DOS. The counter/timer 0 is used for the Systick Timer and is assigned to interrupt level 0. In addition, if the ZT 8980 Super VGA board is placed in the system, then the keyboard interrupt at level 1 is also critical. Should the ZT 8950 Floppy Disk Controller be installed, then interrupt level 6 is required for proper floppy operation. Ziatech's DOS systems use backplane signals INTRQ1* for keyboard interrupts and INTRQ2* for floppy subsystem interrupts. Do not reassign these interrupt selections in DOS systems.

Table 3-5 below lists the pin assignments for the STD bus interrupts.

Table 3-5
STD Bus Interrupts.

Interrupt	Pin Number
INTRQ*	P44
INTRQ1*	P37
INTRQ2*	P50
INTRQ3*	E67



† IR1 and IR2 have other options within the V40. Refer to the OPCN register in Chapter 5 for details.

Figure 3–4. PIC Interrupt Input Requests.

Polled Interrupts on the STD Bus

The PIC can be programmed to supply a unique vector for each of these interrupt inputs. This means only one STD bus interrupt per request can be uniquely defined, as shown in Figure 3-5.

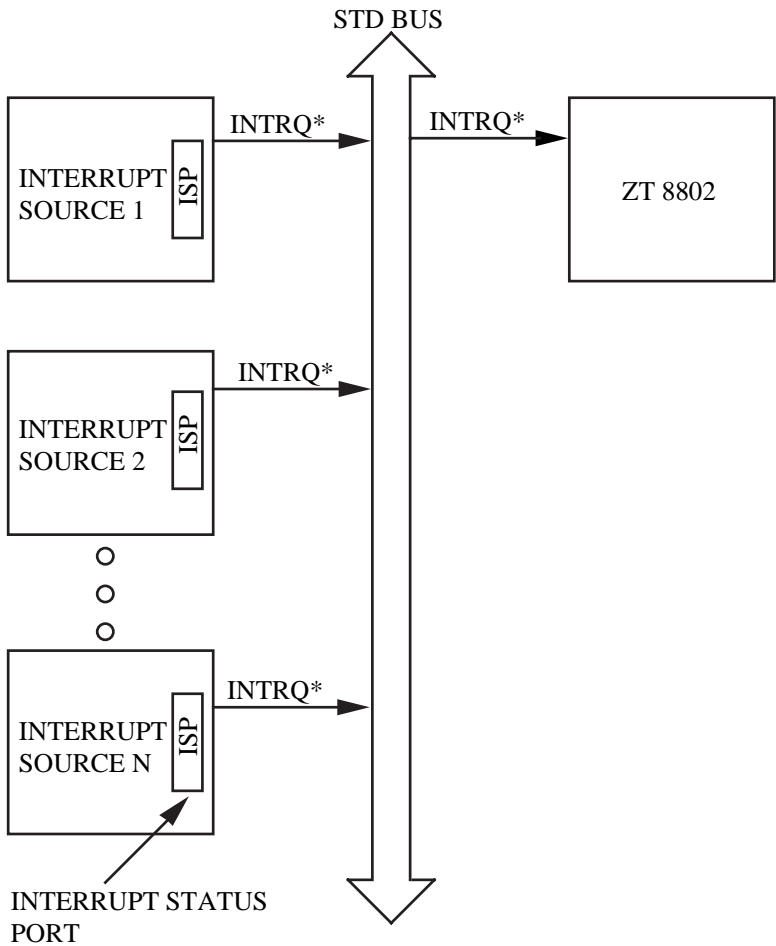


Figure 3-5. Polled Interrupt Structure.

Since DOS expects the use of INTRQ1* and INTRQ2* for particular I/O devices, this leaves only the one INTRQ* signal for all remaining I/O devices in the system. The STD 32 interrupt INTRQ3* is also supported. These devices may share this INTRQ* signal, in which case the application program must poll each possible source to determine which device generated the interrupt. Such a procedure is acceptable for most applications, provided each interrupt source can be polled. A software priority can be established by polling the sources in a specific order.

In general, sharing interrupts requires the use of level-triggered interrupt sources. In this way, if one interrupt exists and another arrives, the level remains active even when the first request has been removed. The level continues to activate the request at the interrupt controller. Conversely, if edge-triggered interrupts are used, the activation of the interrupt request by one device essentially masks the edge created by the second device and the interrupt is not seen by the interrupt controller even though the active logic level remains.

The 8259A PIC can be programmed for either all level-triggered or all edge-triggered interrupts. The use of the counter/timer 0 for the system clock tick requires that the PIC be programmed for edge-triggered interrupts, which is standard for DOS systems. Sharing INTRQ* among two or more devices is not possible under edge-triggered operation.

STD Bus Vectored Interrupts

For more demanding applications, it may be necessary to support each STD bus interrupt source with a unique vector, as illustrated in Figure 3-6. In this mode, up to eight interrupting devices are automatically provided a vector by the on-board 8259 PIC of the ZT 8802. DOS uses the PIC in this mode. Note that the interrupting devices may be on-board as well as off-board.

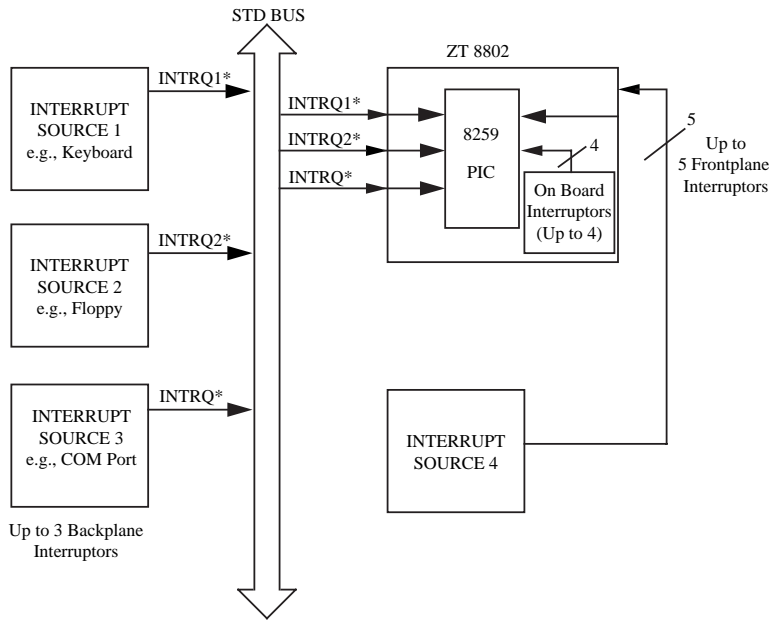


Figure 3-6. Small Scale Vectored Structure.

STD Bus Cascaded Interrupts

To allow for a greater number of interrupts, additional interrupt controllers can be added to the STD bus system, permitting each interrupt source to generate a unique vector for its service routine. The ZT 8802 supports the STD-80 implementation of cascaded interrupt controllers; this is useful for demanding applications that have a large number of interrupt sources. The system is illustrated in Figure 3-7.

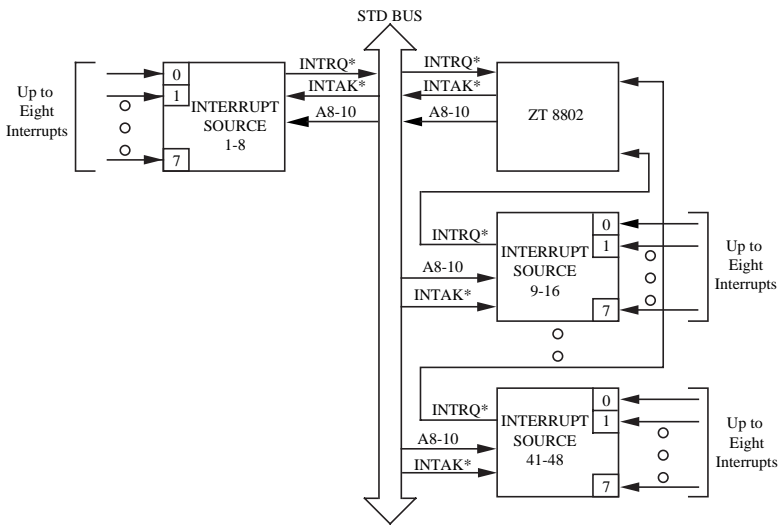


Figure 3-7. Cascaded Interrupt Structure.

Backplane interrupts INTRQ*, INTRQ1*, INTRQ2*, and INTRQ3* and any of the frontplane interrupts may be used to cascade interrupt controllers. The interrupt output from the "slave" interrupt controller in the system is tied to the interrupt request input on the ZT 8802 "master" interrupt controller and the PIC programmed accordingly. Then if an interrupt request from a cascaded interrupt controller is to be serviced, the ZT 8802 drives a 3-bit address known as the cascade address onto A8 through A10 of the STD bus during the interrupt acknowledge cycle. This 3-bit address selects one of the cascaded interrupt controllers to provide an interrupt vector for the requesting input. This interrupt scheme supports a maximum of 56 interrupts from external peripherals using IR1-IR7 of the ZT 8802's master PIC. IR0 is not useable for cascade operation and is dedicated to Timer 0. For a complete discussion of cascaded interrupts, refer to Intel Corporation's application note *AP-59*.

Non-Maskable Interrupts

In addition to the eight interrupt inputs at the interrupt controller, the ZT 8802 supports one source of an interrupt referred to as a "non-maskable" interrupt. This type of interrupt has higher priority over any of the maskable interrupts from the interrupt controller and is not software maskable. The NMIRQ* signal from the STD bus is buffered and presented to the V40 for service.

RESET

The ZT 8802 is equipped with a System Reset circuit, which asserts the STD bus SYSRESET* signal at any time DC voltage is less than 4.75 V. It also drives the SYSRESET* signal during the time a pushbutton switch drives the PBRESET* STD bus signal to the ZT 8802. The pushbutton switch is first debounced on the ZT 8802 before causing a system reset.

During a reset, the CPU, serial ports, and parallel I/O are reset to initial states (these states are detailed in the descriptions of each of these devices later in this manual). The SBX expansion module is also issued a reset. No other devices are affected by the system reset. The SYSRESET* signal is driven to a logical low for a minimum of 250 ms on power up after the power supply reaches 4.75 V (typically 600 ms, max. 1000 ms). Also on power up, the STD bus DCPDN* signal (pin 5) is monotonically driven to a logical 0 until power is at 4.75 V. If power falls below 4.75 V, DCPDN* is again driven to a logical 0. This signal may be used by external boards protecting their RAM during power fail.

CONTROL PORT

The ZT 8802 utilizes eight bits of the 48-point parallel I/O interface for software configurability. The sixth I/O port (FA05h) is used for the control port. If all 48 points of I/O are needed, then you can trade off features by reconfiguring jumpers and bypassing the software configurability. STD ROM users have the board optimized for maximum I/O. The items controlled by these eight bits are memory mode selection, LED, watchdog strobe, real-time clock reset, and timer and SBX oscillator disabling. Figure 3-8 on page 3-32 diagrams the software bit assignments for these features.

In order to avoid read-modify-write errors, we recommend you disable interrupts when modifying this register. The default at power up is logical 0 for all bits, as shown in Figure 3-8 below.

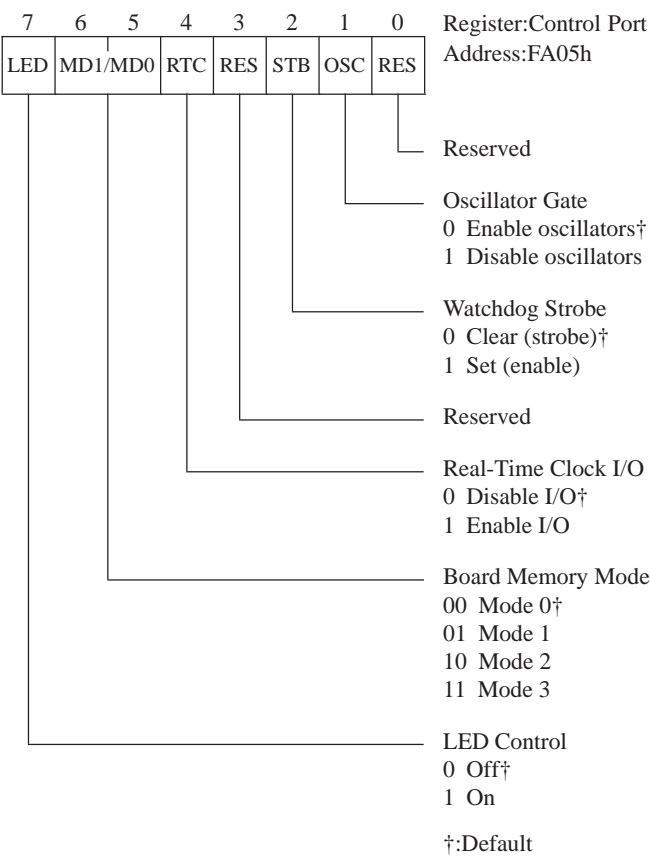


Figure 3–8. Control Port Bit Map.

Hardwiring Features vs. Software Control

All of the features above may be hardwired via jumpers to allow for 48 lines of true off-board I/O. If you are using STD ROM, the board is shipped in this configuration. Figure 3-9 shows the options for these I/O points.

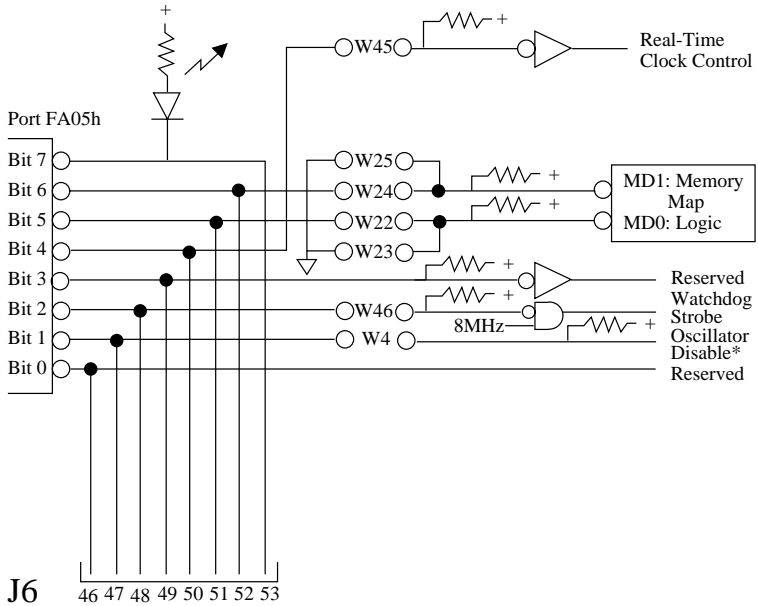


Figure 3-9. I/O Point Jumper Options.

POWER-FAIL PROTECTION

The ZT 8802 supports DC power-fail protection. The advantages of this are described in this section.

DC Power Fail

The factory default setting enables the ZT 8802 to detect 5 VDC and assert a system reset if power falls below 4.75 VDC. In addition, the RAM and Flash memory chip enables are disabled below 4.75 VDC to prevent erroneous data corruption to these devices during power-down/power-up sequences. If the optional battery is installed, battery voltage switches in to protect those circuits selected by jumpers for battery backup. The real-time clock is always protected if the battery is installed. Jumpers allow additional battery backup of the RAM sockets.

DC power-fail protection provides the advantages of system integrity and battery backup. System integrity is improved because the system is not allowed to operate if DC voltage is less than 4.75 V. System reset is held active both on power rising and on power falling; therefore, the processor will not try to function when power is invalid and possibly corrupt valid data in battery-backed devices. DC power-fail detection is important for battery backup so that battery power switches in at the proper time as DC fails.

BATTERY

The ZT 8802 contains a socket for an optional 1 Amp-hour, 3.6 V lithium battery (Ziatech part #BAT-00003). The battery protects the real-time clock in case of power failure and jumpers allow the RAM sockets to also be battery-backed. These jumpers are W33 installed, W32 removed to battery back the RAM socket at 4F, and jumpers W29 installed, W28 removed to battery back the RAM socket at 3F.

Battery life depends on the current requirement per device powered by the battery. Battery life is calculated for both typical and worst-case situations in the following equations. These equations show how battery life decreases with a greater current load. Note that the battery capacity derates at high temperatures.

1. Real-time clock only

a) Typical Data Retention Time (25° C):

Typical Current Drain of Real-Time Clock = 100 nA

$$\begin{aligned}\text{Battery Life (Hours)} &= \text{Battery Capacity/Current Requirements} \\ &= 1 \text{ A}\cdot\text{h}/100 \times 10^{-9} \\ &= 10^7 \text{ Hours}\end{aligned}$$

which is beyond the 10-year shelf life of the battery (87,000 Hours).

b) Minimum Data Retention Time (65° C)

$$\begin{aligned}\text{Battery Life (Hours)} &= \text{Battery Capacity/Current} \\ &= 1 \text{ A}\cdot\text{h}/1 \times 10^{-6}\text{A} \\ &= 10^6 \text{ Hours}\end{aligned}$$

which is limited to the 10-year shelf life of the battery (87,000 Hours).

2. Real-time clock, two 128 Kbyte static RAMs (monolithic Sony part assumed)

- a) Typical Data Retention Time (25° C):

$$\begin{aligned}\text{Total Current} &= \text{RTC} + 2 \times \text{Sony 581000} - 10\text{L} \\ &= 100 \text{ nA} + 2\mu\text{A} = 2.1\mu\text{A}\end{aligned}$$

$$\begin{aligned}\text{Battery Life} &= \text{Battery Capacity/Current} \\ &= 1 \text{ A}\cdot\text{h}/2.1 \times 10^{-6}\text{A} \\ &= 476,000 \text{ hours}\end{aligned}$$

which derates at 25° C to 360,000 hours, still greater than the 10-year shelf life.

- b) Worst Case Retention Time (65° C) with Sony 128K, not low-power:

$$\begin{aligned}\text{Total Current} &= \text{RTC} + 2 \times \text{Sony 581000} - 10\text{L} \\ &= 1 \mu\text{A} + 200 \mu\text{A} \\ &= 201 \mu\text{A}\end{aligned}$$

$$\begin{aligned}\text{Battery Life} &= \text{Battery Capacity/Current} \\ &= 1 \text{ A}\cdot\text{h}/201 \times 10^{-6} \\ &= 4,975 \text{ Hours}\end{aligned}$$

which derates to (4,975)(.87) = 4,328 Hours or .5 Years

- c) Worst Case Retention Time (65° C) with Sony 128K, low-power:

$$\begin{aligned}\text{Total Current} &= \text{RTC} + 2 \times \text{Sony 581000} - 10\text{LL} \\ &= 1 \mu\text{A} + 40 \mu\text{A} \\ &= 41 \mu\text{A}\end{aligned}$$

$$\begin{aligned}\text{Battery Life} &= \text{Battery Capacity/Current} \\ &= 1 \text{ A}\cdot\text{h}/41 \times 10^{-6} \\ &= 24,390 \text{ Hours}\end{aligned}$$

which derates to (24,390)(.87) = 21,220 Hours or 2.4 Years

STATUS INDICATOR (LED)

The ZT 8802 has an on-board LED, located near the extractor, for general purpose use. It is turned on by writing a logical 1 to bit 7 of the Control register at I/O address FA05h, and it is turned off by writing a logical 0 to bit 7 of the same address. See the section on the Control Port, page 3-31, for further detail.

CMOS VERSION OF THE ZT 8802

The ZT 8802 processor board is also available in a CMOS version, the ZT 88CT02. This version provides lower power and extended temperature operation. The differences between the CMOS version and the non-CMOS version with respect to system level issues are discussed briefly here and in further detail in Appendix B, "Specifications," and Appendix C, "PIA System Setup Considerations."

Added Features

The ZT 88CT02 extends the operating temperature range to between -40° and +85° Celsius. All on-board logic is CMOS and utilizes an advanced speed TTL compatible CMOS logic family (ACT) to allow the support of both CMOS and non-CMOS STD bus boards. This is an advantage over strictly CMOS logic (HC or AC), which must receive only CMOS logic levels on its inputs and is therefore restricted to operation with CMOS boards only. The ZT 88CT02 does not have this restriction and is named as such to distinguish both CMOS and TTL compatibility.

In addition to these environmental and electrical advantages, the ZT 88CT02 contains an added feature to support very low power applications. Two of the on-board oscillators may be turned off (timer tick and the SBX oscillator) to conserve power. The V40 may also execute the halt instruction to go into a lower power mode.

Clock Shutdown

Power consumption for CMOS logic is directly proportional to the switching speed of the device. The higher the clock frequency, the greater the power dissipation. In order to minimize the power consumption on the ZT 88CT02 board, the Clock Shutdown feature has been included to allow shutting down the timer tick (1.19318 MHz) and SBX (10 MHz) oscillators. This is dynamically done by writing a 1 to the Control Port at FA05h, bit 1, when W43 is installed. Note that the 16 MHz oscillator that drives the V40 may not be shut down.

Halt with Interrupt Restart

To further decrease power consumption from the Clock Shutdown mode described above, the processor may be halted when processing is not needed and restarted by an interrupt. This interrupt may be from an external source, such as an event requiring service from the processor, or from one of the on-board timers or event sense I/O. If the clock shutdown feature above is not used, the timers continue to run at their full 1.19 MHz frequency and are not affected by a halted processor. Alternatively, the timer clock inputs could count external events and interrupt the processor upon reaching a predetermined count. Any counters not initialized remain idle and do not affect power consumption.

Chapter 4

PROCESSOR DESCRIPTION (V40)

Contents	Page
OVERVIEW	4-2
ZT 8802 Specifics	4-3
FUNCTIONAL BLOCKS	4-4
CPU - Central Processing Unit	4-6
BIU - Bus Interface Unit	4-22
BAU - Bus Arbitration Unit	4-23
CGU - Clock Generator Unit	4-23
VCR - V40 Configuration Registers	4-24
WCU - Wait Control Unit	4-24
SCU - Serial Control Unit	4-24
TCU - Counter/Timer Control Unit	4-25
ICU - Interrupt Control Unit	4-25
DCU - DMA Control Unit	4-25
RESET	4-26
MEMORY AND I/O ADDRESSING	4-27
INTERRUPTS	4-30
Divide Error	4-33
Single-Step	4-34
Non-Maskable	4-34
Fixed Vector Instruction	4-34
Overflow	4-35
Check Index	4-35
Variable Vector Instruction	4-35
Emulation Mode	4-35
8080 EMULATION	4-36

OVERVIEW

The NEC 70208, commonly known as the V40, is a CMOS microprocessor with a 16-bit internal and 8-bit external data bus structure. The V40 instruction set includes all of the instructions of the 8088 and 80188 microprocessors, plus a few more. The added instructions include string I/O, expanded rotate and shift, bit and nibble manipulation, BCD arithmetic, and 8080 emulation mode.

The V40 contains several peripherals frequently used in STD bus applications. These peripherals include a serial controller, interrupt controller, direct memory access (DMA) controller, counter/timers, and a programmable wait-state generator.

This chapter divides the V40 microprocessor into functional blocks and presents an overview of each. More detailed descriptions of the programmable functional blocks are found in subsequent chapters.

ZT 8802 Specifics

The V40 includes a dynamic RAM (DRAM) refresh controller for applications supporting DRAM devices. The ZT 8802 contains only static RAM, so the refresh controller is not needed.

The V40 includes four DMA channels. The ZT 8802 makes use of one of these channels to provide access to on-board memory from external bus masters via the BUSRQ*/BUSAK* sequences. These two STD bus signals are used by external bus masters (such as an off-board DMA controller) to gain control of the STD bus from the ZT 8802. BUSRQ* is driven by the external master to request the bus. The ZT 8802 routes this signal into DMA Channel 0, which if programmed in cascade mode, drives the signal BUSAK* to the external master when the V40 has relinquished control of the on-board memory. The remaining three channels are not supported by the ZT 8802.

FUNCTIONAL BLOCKS

The V40 can be divided into the major functional blocks listed below and shown in Figure 4-1.

CPU	Central Processing Unit
BIU	Bus Interface Unit
BAU	Bus Arbitration Unit
CGU	Clock Generator Unit
VCR	V40 Configuration Registers
WCU	Wait Control Unit
SCU	Serial Control Unit
TCU	Counter/Timer Control Unit
ICU	Interrupt Control Unit
DCU	DMA Control Unit

The following pages discuss each functional block in detail.

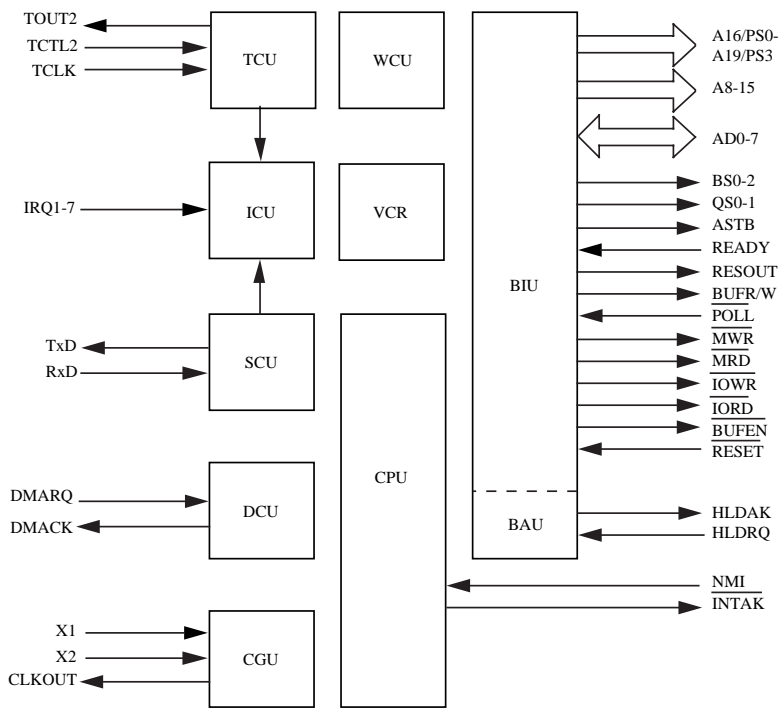


Figure 4-1. V40 Block Diagram.

CPU - Central Processing Unit

The architecture of the CPU functional block is compatible with the 8088. The CPU recognizes all of the instructions found in the 8088 and 80188 microprocessors. Figure 4-2 shows a block diagram of the CPU divided into two elements, the Bus Control Unit (BCU) and the Execution Unit (EXU). The BCU prefetches instructions and data into a 4-byte instruction queue. The EXU executes the instructions. This pipelined architecture increases the throughput over the typical microprocessor that must wait for an instruction or operand to be fetched before operation is continued.

CPU Functional Blocks

The functional blocks in Figure 4-2 are described below. The NEC mnemonic is shown for each block, followed by the Intel mnemonic in brackets. For example, the CPU Flag register is represented by PSW [FL] because NEC labels it Processor Status Word and Intel labels it Flags.

Segment Registers

PS [CS], SS [SS], DS0 [DS], and DS1 [ES]

The CPU can address up to 1 Mbyte of memory in segments of 64 Kbytes or less. The starting address of a segment is specified in a segment register. The four segment registers are as follows:

- PS [CS] - Program Segment register
- SS [SS] - Stack Segment register
- DS0 [DS] - Data Segment register 0
- DS1 [ES] - Data Segment register 1

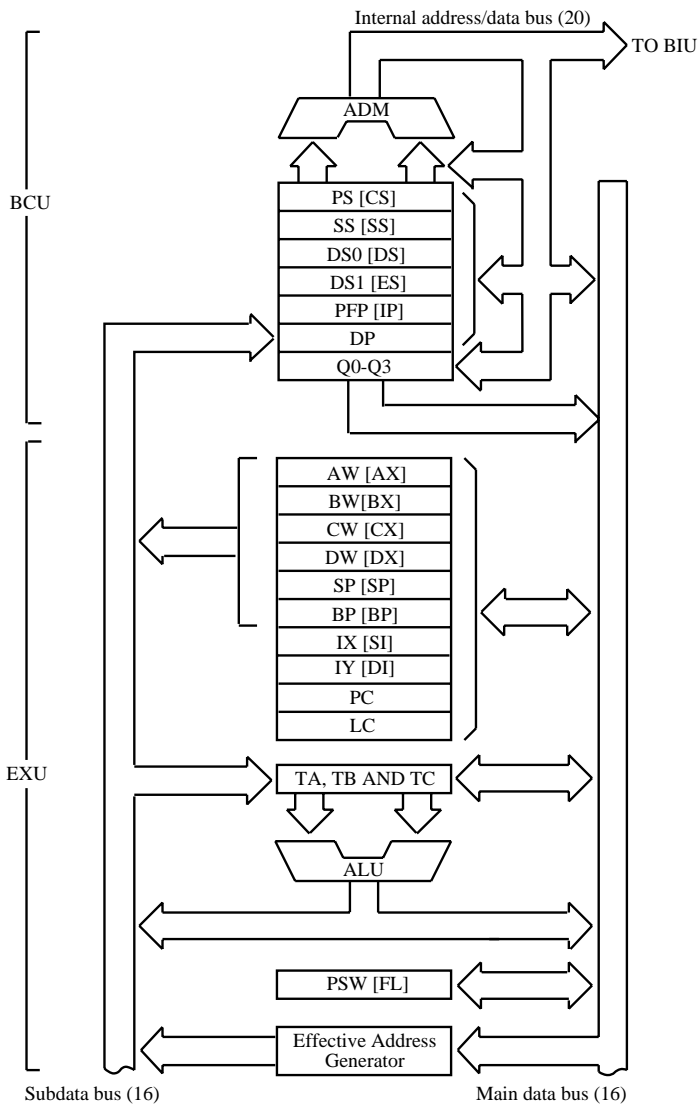


Figure 4-2. CPU Block Diagram.

All memory addresses are specified with a segment and offset as shown in Table 4-1 below. The segment and offset used depend on the type of instruction being executed.

The program always resides in a program segment pointed to by the PS [CS] register. The PFP [IP] always contains the offset within the program segment. The program stack always resides in the stack segment pointed to by the SS [SS] register. The SP [SP] contains the offset of the top of the stack. Stack variables can be addressed using the BP [BP] register because the default segment register is SS [SS].

Table 4-1
Segment Registers.

Memory Reference	Default Segment	Alternate Segment	Offset
Instruction Fetch	PS[CS]	NONE	PFP[IP]
Stack Operation	SS[SS]	NONE	SP[SP]
Variable (except following)	DS0[DS]	PS[CS], DS1[ES], SS[SS]	Effective Address
String Source	DS0[DS]	PS[CS], DS1[ES], SS[SS]	IX[SI]
String Destination	DS1[ES]	NONE	IY[DI]
BP[BP] Used As Base Register	SS[SS]	PS[CS], DS0[DS], DS1[ES]	Effective Address
BW[BX] Used As Base Register	DS0[DS]	PS[CS], DS1[ES], SS[SS]	Effective Address

Program variables generally reside in the data segment with the segment address in the DS0 [DS] register. The offset of the variable within DS0 [DS] is called the effective address. The EXU calculates the effective address by summing any combination of displacement, base register, and index register. The possible combinations of offset, base, and index provide the programmer with a large variety of addressing modes.

Strings are addressed differently from other variables. The segment register used to point to the source string is DS0 [DS] unless an override is used. The offset for the string source is the IX [SI] register. The segment register for the string destination is always DS1 [ES] and the offset is specified in IY [DI].

Prefetch Pointer

PFP [IP]

The prefetch pointer PFP [IP] is a 16-bit binary counter that maintains the offset of the next instruction to be fetched into the instruction queue. The BCU fetches a program instruction based on the segment value in the PS [CS] register and the offset in the PFP [IP].

For sequentially addressed instructions, the PFP [IP] is incremented by the number of bytes of the current instruction to point to the next. For program branching, such as intrasegment and intersegment jumps, the PFP [IP] is programmed with a value contained within the jump instruction. The PFP [IP] is not accessible to the programmer.

Data Pointer

DP

This 16-bit register is the destination for the offset address calculated by the effective address generator. The offset address calculation is done by hardware instead of the traditional microcode, saving three to ten clock cycles for every calculation. The DP register is not accessible to the programmer.

Instruction Queue

Q0 - Q3

The instruction queue is a temporary storage location for program instructions and variables that have been fetched by the BCU to be executed by the EXU. The instruction queue consists of four 8-bit registers, Q0 through Q3. These registers allow instruction fetching by the BCU and instruction execution by the EXU to be independent operations. This overlap essentially eliminates the time required to fetch program instructions and data. Q0 through Q3 are not accessible to the programmer.

Address Modifier

ADM

The V40 uses a 20-bit memory address to access any location in the 1 Mbyte addressing range. The 20-bit memory address is the sum of a segment (shifted left four bits) and an offset. The offset is taken from the PFP [IP] if a program instruction is being addressed or from the DP for all other data. The ADM does this addition. If the PFP [IP] was used, the ADM increments it for the next instruction. The ADM is not accessible to the programmer.

General Purpose Registers

AW [AX], BW [BX], CW [CX], and DW [DX]

The CPU has four 16-bit general purpose registers. Each of these registers can be addressed as one 16-bit register or two 8-bit registers. The 16-bit registers are referred to as AW [AX], BW [BX], CW [CX], and DW [DX]. The high order bytes of the 16-bit registers are AH, BH, CH, and DH, while the low order bytes are AL, BL, CL, and DL. The most common use of these registers is to provide a temporary storage location for data. Some instructions do assign specific meanings to the general purpose registers, as shown in Table 4-2.

Table 4-2
Implied Use of General Registers.

Register	Implied Use	Register	Implied Use
AW [AX]:	Word Multiplication/ Division Word Input/Output	BW [BX]:	Translation
AL:	Byte Multiplication/ Division Byte Input/Output Translation BCD and Decimal Arithmetic	CW [CX]:	String Operations
AH:	Byte Multiplication/ Division	CL:	Variable Shift and Rotate
		DW [DX]:	Word Multiplication/ Division Indirect Input/Output

Pointers and Index Registers

SP [SP], BP [BP], and IX [SI], IY [DI]

The two 16-bit pointer registers are used primarily for stack operations. The Stack Pointer (SP [SP]) is the offset to the top of the stack within the stack segment. This pointer is adjusted automatically each time a stack operation is performed. The Base Pointer (BP [BP]) is an offset to any location within the stack segment. The BP [BP] is useful as a pointer to variables being passed on the stack. Both pointer registers are accessible to the programmer.

The 16-bit index registers are used primarily for string operations. Strings are linear arrays of data that can be organized as words, bytes, nibbles, or even bit values. The index registers specify the offset of the string source (IX [SI]) and destination (IY [DI]) within Data Segment 0 (DS0 [DS]) and Data Segment 1 (DS1 [ES]), respectively. The index registers are adjusted automatically during string transfers. Both IX [SI] and IY [DI] are accessible to the programmer.

Program Counter

PC

The program counter is a 16-bit binary counter that contains the offset address of the next instruction to be executed by the EXU. The PC is automatically incremented each time the EXU reads an instruction from the queue. If the instruction causes a branch in program execution, the PC is programmed with the branch address. At this point the contents of the PC are the same as the PFP [IP]. The difference between the PFP [IP] and the PC is the PFP [IP] contains the offset of the next instruction to be fetched by the BCU and the PC contains the address of the next instruction to be executed by the EXU. The PC is not accessible to the programmer.

Loop Counter

LC

LC is a binary counter used to regulate iterative operations such as string transfers controlled by the repeat prefix and multiple-bit shifts and rotations. The CPU uses hardware for a loop counter as opposed to microcode used by the traditional microprocessor.

Temporary Registers A, B, and C

TA, TB, and TC

These 16-bit registers are used by the Arithmetic and Logic Unit (ALU) during arithmetic and logical instructions such as multiplication, division, and shift and rotate. TA and TB combine for 32-bit temporary storage during multiplication and division. The programmer does not have access to the temporary registers.

Arithmetic and Logic Unit

ALU

The ALU performs arithmetic and logic operations as well as bit manipulation. Arithmetic and logic operations include add, subtract, multiply, divide, increment, decrement, compare, complement, AND, OR, and exclusive OR. Bit manipulation includes shifting, rotating, comparing, setting, clearing, and inverting of individual bits.

Effective Address Generator

EAG

The 16-bit offset address calculated by the EXU for memory operations is called the effective address. The effective address can include a displacement, base, index, or combination of the three, depending on the addressing mode specified in the instruction being executed. Traditional microprocessors calculate this effective address using microcode. The EAG does this calculation in hardware. As an example, the 8088 requires up to 12 clocks to calculate the effective address using microcode. However, the V40 does all effective address calculations in two clocks with the hardware EAG.

The effective address, once calculated by the EAG, is transferred to the DP register where it can be used by the BCU to transfer data between the CPU and memory.

Processor Status Word

PSW [FL]

There are six status flags and four control flags in the 16-bit PSW [FL], as seen in Figure 4-3. Notice that not all 16 bits are defined. Those not defined are reserved; that is, they may be used in later versions of the processor. Because of this, a program should never rely on a value in any of these reserved bits.

The status flags provide information about the result of arithmetic and logic operations. The status flags are set (logical 1) and reset (logical 0) by the EXU based on the result of an arithmetic or logic operation. These flags can be tested by conditional jump instructions to change the order of program execution.

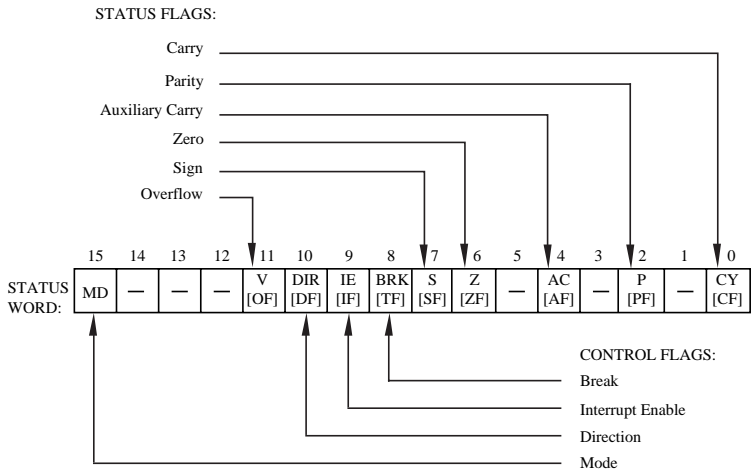


Figure 4–3. Processor Status Word.

The control flags are used by the programmer to direct CPU operation. The control flags are set (logical 1) and reset (logical 0) with dedicated instructions. The IE [IF] and BRK [TF] flags are automatically reset when the program enters an interrupt service routine.

The PSW [FL] is automatically preserved on the stack at the start of an interrupt service routine for both hardware- and software-initiated interrupts and at the start of a procedure initiated with the CALL instruction. A return from interrupt instruction, a return from procedure instruction, or a return from emulation instruction restores the contents of the PSW [FL] from the stack.

Different instructions affect the status flags differently. A detailed description of each status flag is given on the following pages. Reference to bit position is based on the least significant bit being bit 0. The state of a flag is referred to as "set" when a logical 1 is present and "reset" when a logical 0 is present.

Status Flags

CY [CF] (Carry Flag)

The carry flag is set if an addition results in a carry out of bit 7 for byte operations or bit 15 for word operations. The CY [CF] flag is also set if a subtraction results in a borrow into bit 7 for byte operation or bit 15 for word operations.

For unsigned byte multiplication, CY [CF] is reset if the most significant byte of the result (register AH) is 0. The same is true of the most significant word (register DW [DX]) for unsigned word multiplication.

For signed multiplication, CY [CF] is reset if the sign bit of the least significant byte (register AL) is extended to the most significant byte (register AH). The same is true for signed word multiplication with the least significant word in register AX and the most significant in register DX.

P [PF] (Parity Flag)

The P [PF] flag is set if the least significant byte of an arithmetic or logical result has an even number of bits set. This flag is useful for checking the parity of ASCII characters.

AC [AF] (Auxiliary Flag)

AC [AF] is set if an addition results in a carry from the four least significant bits of the result. This is true for both byte and word addition. This flag is used by the CPU for BCD arithmetic operations.

Z [ZF] (Zero Flag)

Z [ZF] is set if the result of an arithmetic or logical operation is zero. A common use of this flag is to determine if two numbers are equal. The program subtracts the two values and if Z [ZF] is set, the values are equal.

S [SF] (Sign Flag)

Arithmetic and logic operations set S [SF] equal to the high order bit of the result. This is bit 7 for byte operations and bit 15 for word operations. For signed binary operations, S [SF] is reset for positive results and set for negative results. Programs using unsigned operations usually ignore S [SF] because the high order bit does not reflect the sign of the result.

V [OF] (Overflow Flag)

The V [OF] flag is set if the result of an operation is a positive number that is too large or a negative number that is too small to fit into the destination. The application program can use the overflow flag to determine if the result of two's complement arithmetic operation is out of range.

Control Flags

MD (Mode Flag)

The CPU operates in either native or emulation mode. In native mode, the CPU executes the standard 8086/186 instructions in addition to instructions unique to the V40. In emulation mode, the CPU executes an 8080 based instruction set. The MD flag is used to distinguish between the two modes. MD is programmed using specific instructions to put the CPU in the native mode (MD is set) or emulation mode (MD is reset). Refer to page 4-35 for more information.

DIR [DF] (Direction Flag)

The CPU supports string operations to manipulate linear arrays of data organized as words, bytes, nibbles, or bits. Index registers are used to point to elements of the array during a string operation. After a string operation is completed, the index registers are incremented or decremented depending on the state of the DIR [DF] flag. If the DIR [DF] flag is set, the index is incremented to point to the next array element. If the DIR [DF] flag is reset, the index is decremented.

IE [IF] (Interrupt Enable Flag)

The IE [IF] flag determines how the CPU responds to maskable external interrupts. If IE [IF] is set, the CPU recognizes maskable external interrupts. The CPU ignores all maskable external interrupts if IE is reset. IE [IF] has no effect on external non-maskable interrupts or internally generated interrupts. IE [IF] is set or reset with dedicated instructions, but will also be reset automatically with a return from interrupt instruction.

BRK [TF] (Break Flag)

Setting the break (or trap) flag puts the CPU into a single-step operation useful for testing program execution. With BRK [TF] set, the CPU automatically generates an internal interrupt after each instruction. The programmer need only develop an interrupt service routine to examine contents of registers, dump memory, or do whatever is necessary for testing.

The BRK [TF] flag is set or reset by transferring the PSW [FL] to the program stack and using memory manipulation instructions to modify it. Once BRK [TF] is modified, it must be transferred back to the PSW [FL] to generate a type 1 interrupt after the execution of each instruction. As part of the interrupt acknowledge, the PSW [FL] is saved on the stack and the BRK [TF] flag is reset. This is done so the processor will not single-step through the interrupt service routine. Once the service routine is completed, the PSW [FL] is restored from the stack automatically, setting the BRK [TF] flag to trap the next instruction.

Enhanced Architecture

The V40 CPU includes several enhancements that provide an increase in performance over the 8088 microprocessor found on many STD bus designs. The most noticeable performance improvements come from additional hardware for the Effective Address Generator, Loop Counters and Shifters, and the use of dual internal data buses.

Using a hardware-effective address generator rather than microcode reduces the time to fetch memory operands by as much as 10 clocks per fetch. Using hardware counters and shifters instead of the conventional microcode increases the speed of multiply and divide instructions by as much as four times. Dual internal data buses reduce traffic for instructions with two operands and for effective address calculation. These enhancements add up to a speed increase of as much as 30 percent over the 8088 microprocessor.

Standby Mode

The CPU has a standby mode to reduce power consumption by more than one-tenth during idle periods. Standby mode is automatically entered when the HALT instruction is executed from the native or 8080 emulation mode. This does not affect any of the internal peripherals, such as the counter/timers, interrupt controller, refresh controller, or DMA controller. The CPU automatically exits the standby mode after a reset or an interrupt.

BIU - Bus Interface Unit

The BIU controls the external address, data, and control buses. The BIU also synchronizes the RESET and READY inputs with the clock, as shown in Figure 4-4. The synchronized RESET signal is used internally. It is provided externally as a signal called RESOUT. The synchronized READY signal is combined with the output of the Wait Control Unit to control the number of wait states inserted during bus operations.

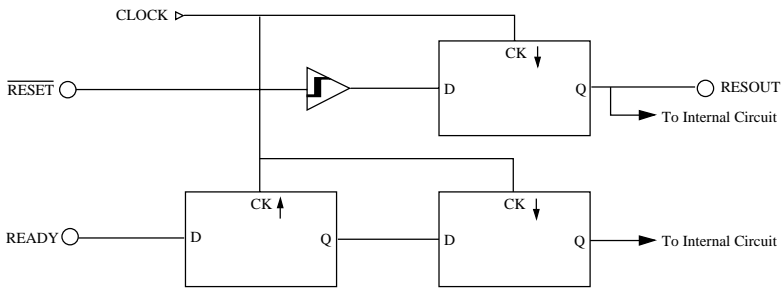


Figure 4-4. RESET and READY Synchronization.

BAU - Bus Arbitration Unit

The V40 includes two internal bus masters and two signals supporting one external bus master. The two internal bus masters are the Central Processing Unit and DMA Control Unit. An external master, such as the 8087 numeric data processor, is supported with the HLDRQ and HLDAK signals. Each of the three bus masters mentioned above needs access to the address, data, and control buses to perform their function. The BAU controls which of these bus masters has access to the buses at any given time. The bus masters are prioritized in the following order:

- (1st) DCU - DMA Control Unit
- (2nd) HLDRQ - External Bus Master
- (3rd) CPU - Central Processing Unit

If one bus master is using the bus and another bus master with higher priority makes a request, the BAU inactivates the current bus master's acknowledge. The BAU grants access to the higher priority bus master after the current bus master removes the request.

The BUSLOCK prefix prevents all bus masters other than the CPU from gaining access to the bus.

CGU - Clock Generator Unit

The CGU halves the frequency of the external oscillator to provide a clock reference with a 50% duty cycle to the CPU. This same signal is available on an external pin called CLKOUT to which all of the V40 timing parameters are referenced.

VCR - V40 Configuration Registers

Twelve programmable registers are used to configure the V40 to meet the needs of varying applications. The V40 configuration registers are located in the top 16 bytes of the 64 Kbytes of I/O address space. The configuration registers define the functions of the programmable pins; the enabling and disabling of the SCU, TCU, ICU, and DCU; the location of the SCU, TCU, ICU, and DCU programmable registers in I/O address space; the wait-state configuration; DRAM refresh; and the counter/timer clock source.

WCU - Wait Control Unit

The WCU provides added flexibility for interfacing to memory and I/O that have varying speed requirements. The V40 includes three internal bus masters that access memory and I/O devices. The number of wait states inserted can be programmed separately for the CPU, RCU, and DCU. The memory space can be divided into three separate areas and the number of wait states defined differently for each. The wait states are programmed through the WCY2, WCY1, and WMB V40 configuration registers.

SCU - Serial Control Unit

The serial control unit is a single asynchronous serial channel used for serial communication between the V40 and a serial device external to the V40. Programming the SCU is similar to programming the 8251A Serial Control Unit for "asynchronous" modes of operation.

TCU - Counter/Timer Control Unit

The Counter/Timer Control Unit includes three 16-bit programmable counter/timers. These counter/timers can be used for SCU baud rate generation, timing loops, timed and periodic interrupts, and external asynchronous event counters. Programming the TCU is similar to programming the 8254 Programmable Interval Timer, with a few restrictions placed on operating modes because of the way the TCU is connected internally to the V40.

ICU - Interrupt Control Unit

Interrupts provide an efficient interface between the V40 CPU and supporting peripheral devices. The ICU supports eight interrupts directly and can be cascaded with other interrupt controllers, such as the 8259A Programmable Interrupt Controller, to support additional interrupting sources. Programming the ICU is similar to programming the 8259A.

DCU - DMA Control Unit

The DCU controls high speed data transfer between I/O and memory devices. The DCU is similar to the 8257 Programmable DMA Controller except the DCU supports the full 1 Mbyte of V40 addressing space. The ZT 8802 supports the DCU in Cascade configuration only.

RESET

Resetting the V40 initializes registers internal to the CPU, VCR, SCU, TCU, ICU, and DCU. The reset states for the CPU registers are given in Table 4-3. The reset states for registers internal to the VCR, TCU, ICU, DCU, and SCU are given in their respective chapters, Chapters 5 through 9.

The reset states of the program segment and instruction pointer combine to produce a physical address of FFFF0h. This is the address from which the V40 fetches the first instruction after reset.

Table 4-3
CPU Reset State.

Register	Reset Value
PFP [IP]	0000h
PC	0000h
PS [CS]	FFFFh
SS [SS]	0000h
DS0 [DS]	0000h
DS1 [ES]	0000h
PSW [FL]	F002h
AW [AX], BW [BX]	Undefined
CW [CX], DW [DX]	Undefined
IX [SI], IY [DI]	Undefined
BP [BP], SP [SP]	Undefined
Instruction Queue	Cleared

MEMORY AND I/O ADDRESSING

This section discusses how the V40 communicates with memory and I/O devices. The V40 has a 20-bit address bus and an 8-bit data bus. With 20 bits of address, the V40 can directly access up to 1 Mbyte of memory. The address range is from 0 to FFFFFh, as shown in Figure 4-5. Address locations 0 to 7Fh are reserved for dedicated interrupts and future enhancements. The address range from 80 to 3FFh completes the interrupt vector table and may be used as needed by the application. The 12 bytes (6 words) from FFFF0 to FFFFBh are the area vectored to by the V40 after a reset. The most common practice is to program this area with an intersegment jump to the start of the application program. The upper four bytes are reserved and must not be programmed.

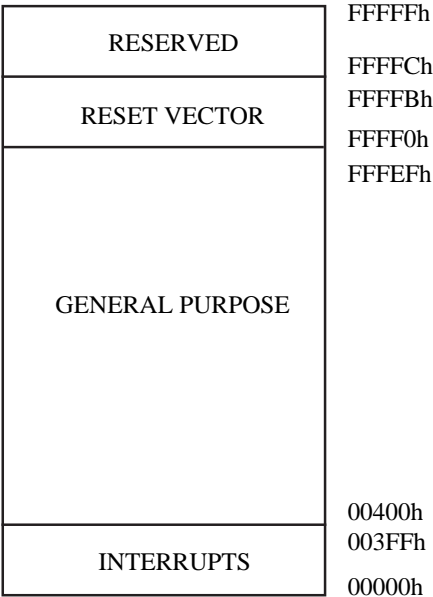


Figure 4–5. Generic V40 Memory Map.

To the programmer, the V40 address space is organized as a contiguous sequence of up to 1 Mbyte. Data can be addressed in units of bytes, words, and double words. Word and double-word values are stored in memory with the most significant byte at the higher address and the least significant at the lower. Figure 4-6 illustrates these data formats.

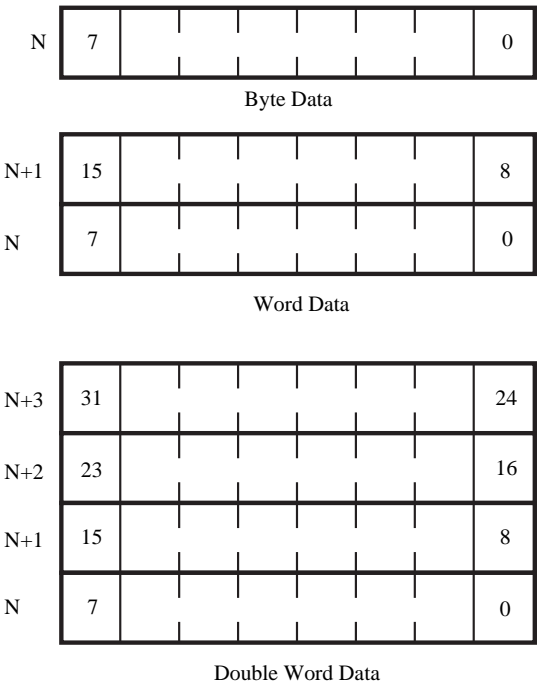


Figure 4-6. Data Formats.

The lower 16 lines of the 20 address lines are also used to address I/O devices. With 16 bits of address, the V40 can directly access up to 64 Kbytes of I/O. The address range is from 0 to FFFFh, as shown in Figure 4-7. Address locations FF00 through FFEFh are reserved for future use. The address range from FFF0 through FFFFh is currently used for the V40 configuration registers. These are registers that define programmable options in the V40, such as wait-state insertion, location of internal peripheral device registers, enabling DRAM refresh, and selecting the period of refresh.

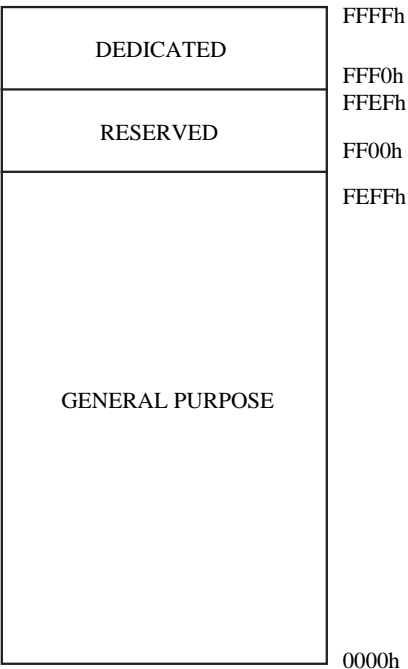


Figure 4-7. Generic V40 I/O Map.

INTERRUPTS

The V40 includes a versatile interrupt structure to support both hardware and software initiated interrupts. Hardware interrupts are external inputs to the V40 and can be classified as maskable or non-maskable. Maskable interrupts are routed to the CPU through the ICU. The ICU provides the maskable interrupt inputs with the ability to be level- or edge-triggered, have fixed or rotating priorities, and be individually masked. Non-maskable interrupts are routed directly to the CPU and are not maskable through programming.

Table 4-4
Interrupt Sources.

	Interrupt Source	Clocks	Priority
Software	DIVU divide error	45	1
	DIV divide error	45-55	1
	CHKIND breakout error	53-56	1
	BRKV	40	1
	BRK3	38	1
	BRK imm8	38	1
	BRKEM imm8	38	1
	CALLN imm8	38	1
	BRK single step	38	4
Hardware	NMI	38	2
	ICU inputs	49	3

Software interrupts are internally generated during program execution, including instructions that can be executed to generate interrupts and error handling for such things as a divide overflow. Table 4-4 lists the sources of hardware and software interrupts. The remainder of this chapter explains these interrupts and how the V40 handles them.

The purpose of an interrupt is to redirect the CPU from its current activity to an interrupt service routine designed to handle the needs of the interrupting source. Every interrupting source is associated with a number that points the CPU to a location in memory that contains the address of the interrupt service routine. This number is called an interrupt vector, and the area in memory where the addresses of the interrupt service routines are stored is called the interrupt vector table. During an interrupt cycle, the CPU multiplies the vector by four to obtain the location of the service routine address in the vector table. The CPU then transfers control to the service routine at the address read from the vector table. This operation is illustrated in Figure 4-8. It is the programmer's responsibility to load the address of the service routine into the vector table. The address includes a segment and offset value in the format shown.

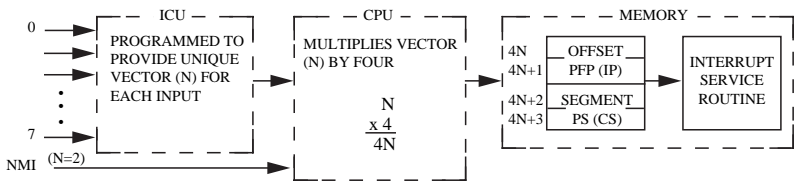


Figure 4-8. Interrupt Processing.

The vector of an interrupt source must be known before the location of the service routine address can be determined. The interrupt vector table, shown in Table 4-5, lists the vectors for the sources of interrupts. For example, assume you need to write a service routine to handle a non-maskable interrupt (NMI) request. The vector for NMI is 2, as seen in Table 4-5. The address of the NMI service routine is determined by the application, but that address must be programmed in the vector table at 8h.

Table 4-5
Interrupt Vector Table.

	Interrupt Source	Vector Number
Software	DIVU divide error	0
	DIV divide error	0
	CHKIND breakout error	5
	BRKV	4
	BRK3	3
	BRK imm8	32-255
	5l BRKEM imm8	32-255
	CALLN imm8	32-255
	BRK single step	1
Hardware	NMI	2
	ICU inputs	32-255

Before describing each source of interrupt shown in the vector table, it is useful to summarize the operation of the CPU in response to an interrupt. Interrupts come to the CPU from three sources: the NMI signal external to the V40, the output of the ICU, and from inside the CPU to itself. A vector is supplied in all cases to distinguish between the interrupting sources. The CPU determines the address of the service routine by multiplying the vector times four. Before transferring execution to the service routine, the CPU saves the machine status by pushing the current contents of the PSW [FL] and the return address onto the stack. The CPU then clears the BRK [TF] and IE [IF] flags to prevent subsequent single-step and maskable interrupts, and transfers program execution to the service routine. The service routine is terminated with a "return from interrupt" instruction. This instruction causes the CPU to restore the PSW [FL] from the stack and return execution to the interrupted program. Restoring the PSW [FL] automatically enables single-step and maskable interrupts.

Divide Error

The divide error interrupt is generated by the CPU following execution of a DIV [DIV] or DIVU [IDIV] instruction if the calculated quotient is larger than the specified destination. The interrupt is not maskable and the vector is fixed at 0.

Single-Step

The single-step interrupt is a powerful software debugging tool. The purpose of the single-step interrupt is for software single stepping through a sequence of code. This interrupt is controlled by the BRK [TF] flag in the PSW [FL]. There is no instruction to set the BRK [TF] flag. To set the BRK [TF] flag, the PSW [FL] register must be pushed on the stack, the flag set, and the PSW [FL] popped back off the stack. With the BRK [TF] flag set, a single-step interrupt is generated after each instruction. The CPU responds to the interrupt by pushing the PSW [FL], PS [CS], and PFP [IP] on the stack. The BRK [TF] and IE [IF] are reset to a logical 0 to prevent another single-step or maskable interrupt. Upon completion of the single-step routine, the CPU restores the PSW [FL], PS [CS], and PSP [IP]. The single-step interrupt is not masked by the IE [IF] bit in the PSW [FL].

Non-Maskable

The V40 has a non-maskable interrupt input called NMI. NMI is rising edge triggered but must remain active for two CPU clocks to guarantee recognition. This interrupt is not maskable and the vector is fixed at 2.

Fixed Vector Instruction

The fixed vector instruction (BRK3 [INT3]) is a special form of the more general variable vector instruction. The difference is the fixed vector is a single byte while the variable vector requires two bytes. The primary use of this instruction is for breakpoint execution during program development. Because it is a single byte, the instruction can be mapped over the smallest possible instruction. This interrupt is not maskable and has a fixed vector of 3.

Overflow

The overflow interrupt is generated if the V [OF] flag is set to a logical 0 and the BRKV [INTO] instruction is executed. This interrupt is useful for trapping overflow errors for mathematical operations. The overflow interrupt is not maskable and the vector is fixed at 4.

Check Index

The purpose of the check index instruction is to test the index of an array against an upper and lower limit. The CHKIND [BOUND] instruction generates a check index interrupt if the index value is less than the lower limit or greater than the upper limit. The vector for the check index instruction is fixed at 5 and is not maskable.

Variable Vector Instruction

Interrupts can be generated using a variable vector interrupt instruction with the format BRK [INT] *xx*, where *xx* is the vector number. Accessing subroutines in this manner means only the subroutine vector is fixed. The location and length of the subroutine can vary without affecting the main program. These interrupts cannot be masked.

Emulation Mode

Two interrupt instructions deal with 8080 emulation mode. The BRKEM instruction is used to transfer V40 operating modes from native to emulation for execution of 8080 based programs. The CALLN instruction is used in emulation mode to call an 8088 procedure. Both instructions have the format and operation of the variable vector instruction. The following pages discuss 8080 emulation in detail.

8080 EMULATION

Designs based on 8080 and 8085 microprocessors have two major limitations: not enough performance and lack of development tools. Upgrading an 8-bit design to a higher performance microprocessor requires time to convert the software. The V40 solves these problems by supporting two modes of operation, emulation and native. When the CPU is in emulation mode, it executes the 8080 instruction set. Emulation mode is used for the existing base of software. In native mode, the CPU executes the 8088 instruction set. All future software development is done in native mode to take advantage of the 8088 instruction set and the large number of development tools designed around it.

The CPU powers up in native mode, which is the normal mode of operation. Two instructions are provided to switch the CPU from native mode to emulation mode and back. Break for Emulation (BRKEM) is the instruction used to switch from native to emulation mode, and Return from Emulation (RETEM) is used to switch back. The effect of these instructions and emulation mode operation is discussed below.

The BRKEM instruction is similar to the BRK [INT] software interrupt. BRKEM includes an 8-bit vector that, when multiplied by four, points to the location in the interrupt vector table that contains the address of the 8080-based routine. During execution of this instruction, the CPU saves the machine status by pushing the current contents of the PSW [FL] and the return address onto the stack. The CPU then clears the mode (MD) flag to a logical 0 and loads the address of the emulation mode routine into the PS [CS] and PFP [IP].

The RETEM instruction is one of four methods to terminate emulation mode. The execution of RETEM is identical in operation to the RETI [IRET] instruction. Upon executing this instruction, the CPU restores the contents of the PSW [FL], PS [CS], and PFP [IP] from the stack, returning program execution to the instruction following BRKEM. The other three methods of exiting emulation mode are a system reset, a hardware interrupt, or the CALLN instruction.

A hardware interrupt suspends the 8080 emulation mode. The CPU pushes the PSW [FL] and return address onto the native mode stack, sets the MD flag to a logical 1, and transfers program execution to the native mode interrupt service routine. When the CPU executes the RETI [IRET] instruction, the PSW [FL] is restored with the MD flag set to a logical 0 and program execution continues in the emulation mode. The CALLN instruction permits the execution of native mode subroutines from emulation mode. The CPU responds to CALLN in the same manner as a hardware interrupt.

The emulation mode cannot be nested. For example, assume the CPU is operating in native mode and executes the BRKEM instruction. The CPU switches to native mode and begins executing emulation code. Next, assume a hardware interrupt (such as the 16450 serial controller) suspends emulation mode and the CPU begins executing the interrupt service routine in native mode. That interrupt service routine cannot include a BRKEM instruction.

Table 4-6 shows the relationship between the native and emulation mode registers and flags. The native mode registers not shown are inaccessible to 8080 programs. They are as follows: AH [AH], PS [CS], SS [SS], DS0 [DS], DS1 [ES], IX [SI], IY [DS], and the upper eight bits of the PSW [FL].

Table 4-6
Emulation Mode Registers and Flags.

	8080	8088
Registers	A	AL
	B	CH
	C	CL
	D	DH
	E	DL
	H	BH
	L	BL
	SP	BP
	PC	PC
Flags	C	CY
	Z	Z
	S	S
	P	P
	AC	AC

Memory addressing and stack referencing must also be considered. The 8080 addresses a maximum of 64 Kbytes. This block of memory can be located anywhere in the 1 Mbyte address space by programming the PS [CS] word in the interrupt vector table before the BRKEM instruction is executed. All data and stack operations are referenced from the DS0 [DS] register. This register must be initialized before the BRKEM instruction is executed. The values in the PS [CS] and DS0 [DS] registers must be equal for complete compatibility with the 8080 structure.

Emulation mode uses the BP [BP] register for the stack pointer instead of the native mode SP [SP] register in order to reduce the possibility of programming errors in one mode corrupting the stack of the other. This feature is helpful during program development.

Chapter 5

PROCESSOR CONFIGURATION (V40)

Contents	Page
OVERVIEW	5-2
VCR - V40 CONFIGURATION REGISTERS	5-2
OPCN - On-Chip Peripheral Connection Register	5-4
OPSEL - On-Chip Peripheral Selection Register	5-5
OPHA, DULA, IULA, TULA, and SULA	5-6
WCY2 - Wait Cycle 2 Register	5-8
WCY1 - Wait Cycle 1 Register	5-8
WMB - Wait Memory Boundary Register	5-10
RFC - Refresh Control Register	5-11
TCKS - Counter/Timer Clock Selection Register	5-12
RESET	5-14

OVERVIEW

The V40 is a high integration microprocessor containing a CPU and several peripherals most commonly found in STD bus systems. The V40 also includes a software programmable register set to configure these peripherals to specific applications. This chapter explains the architecture and use of these configuration registers.

VCR - V40 CONFIGURATION REGISTERS

The V40 has 16 configuration registers (4 are reserved), mapped from I/O address FFF0 through FFFFh. The registers are listed in Table 5-1 and are discussed in detail on the following pages. All of the registers can be written to with the output instruction and read from with the input instruction. *The value input may be different from the value output, but only in the bits not defined.*

Table 5-1
V40 Configuration Registers.

I/O Address	Register	Function
FFFFh	Reserved	--
FFFEh	OPCN	V40 multiplexed pin assignment
FFFDh	OPSEL	V40 peripheral enable
FFFCh	OPHA	V40 peripheral I/O address (MSB)
FFFBh	DULA	DCU I/O address (LSB)
FFFAh	IULA	ICU I/O address (LSB)
FFF9h	TULA	TCU I/O address (LSB)
FFF8h	SULA	SCU I/O address (LSB)
FFF7h	Reserved	--
FFF6h	WCY2	DCU wait states
FFF5h	WCY1	CPU memory and I/O wait states
FFF4h	WMB	Memory wait state boundaries
FFF3h	Reserved	--
FFF2h	RFC	Refresh enable & frequency select
FFF1h	Reserved	--
FFF0h	TCKS	Timer/counter clock selection

OPCN - On-Chip Peripheral Connection Register

The OPCN register is shown in Figure 5-1. This register allows the application to configure the V40 with different capabilities as required.

The two bits of the IRSW field select the interrupt source to be assigned to IRQ1 and IRQ2 of the interrupt controller. The values programmed into bits 2 and 3 depend on the use of the interrupt controller in the application. The pins external to the V40 used for IRQ1 and IRQ2 are connected to the interrupt jumper block.

Bits 0 and 1 are programmed to a 0 and 1 respectively to enable the V40 serial port.

The STD ROM software initializes OPCN to a 06h to configure for the V40 serial port and to use the SCU interrupt on INT1 and IRQ2 on INT2.

The Ziatech DOS system initializes OPCN to a 02h to configure for the V40 serial port and to use IRQ1 for INT1 and IRQ2 for INT2. Note that DOS systems require the keyboard controller interrupt on IRQ1 and the floppy disk controller interrupt on IRQ2.

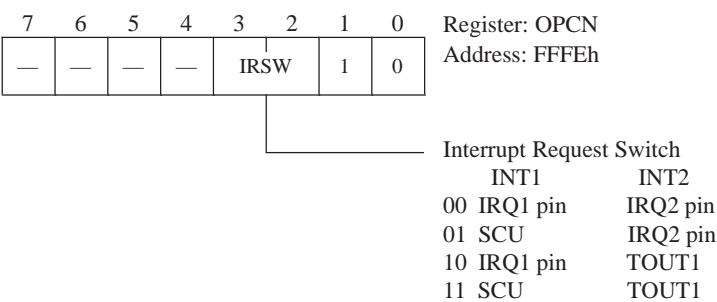


Figure 5-1. On-Chip Peripheral Connection Register.

OPSEL - On-Chip Peripheral Selection Register

The V40 integrates several of the most common peripheral devices with a CPU in one package. The peripheral devices include a serial port, interrupt controller, DMA controller, and three counter/timers. The OPSEL register enables or disables these peripheral devices. The format of the OPSEL register is shown in Figure 5-2. No restrictions are placed on the use of the OPSEL register. The Ziatech DOS system initializes OPSEL to a 0Fh to enable all of the internal peripherals. The STD ROM software initializes OPSEL to a 0Eh to enable the ICU, the V40 serial port, and the counter/timers for baud rate generation.

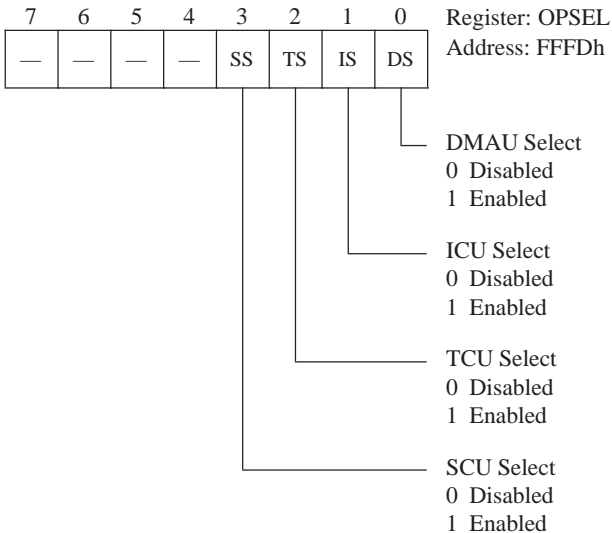


Figure 5-2. On-Chip Peripheral Selection Register.

OPHA, DULA, IULA, TULA, and SULA

Five registers determine the I/O base address of the programmable registers used to communicate with the DMA controller, interrupt controller, timer/counters, and serial controller.

OPHA - On-Chip Peripheral High Address register (FFFC_h)

DULA - DMA Unit Low Address register (FFFB_h)

IULA - Interrupt Unit Low Address register (FFFA_h)

TULA - Timer/Counter Unit Low Address register (FFF9_h)

SULA - Serial Unit Low Address register (FFF8_h)

The I/O base address is a 16-bit value made up of two 8-bit values. The upper eight bits of the address for the DMA controller, interrupt controller, counter/timers, and serial controller are defined by the OPHA register. The lower eight bits for the DMA channel are programmed in the DULA register. The same holds true for the interrupt controller and IULA register, for the counter/timers and TULA register, and for the serial controller and SULA register.

In operation, OPHA permits the four internal peripheral devices to be mapped to any 256-byte block in the 64 Kbyte I/O address space. The individual registers DULA, IULA, TULA, and SULA are programmed to define the base address of each of these devices anywhere within this block. For example, if the interrupt controller is to be mapped starting at I/O address FF20_h, OPHA must be programmed with an FF_h and IULA with a 20_h.

The only restriction placed on the programming of these registers is to be sure the internal V40 peripherals are not mapped in the same address range as other I/O devices local to the ZT 8802. The STD ROM and Ziatech DOS software programs these registers with the values shown in Table 5-2.

Table 5-2
STD ROM and Ziatech DOS Address Selection.

Register	Value	I/O Port Address
OPHA	00	-----
DULA	D0	00D0
IULA	20	0020
TULA	40	0040
SULA	B0	00B0

WCY2 - Wait Cycle 2 Register

The V40 includes a programmable wait-state generator to interface to memory and I/O devices that are not fast enough to operate without wait states. The wait-state generator is programmed through the WCY2, WCY1, and WMB configuration registers. The format of the WCY2 register is shown in Figure 5-3. The STD ROM software initializes the WCY2 register with a 08h. The Ziatech DOS system programs the WCY2 register with a 00h.

7	6	5	4	3	2	1	0	Register:WCY2
—	—	—	—	1/0	0	—	—	Address:FFF6h

Figure 5–3. Wait-Cycle 2 Register.

WCY1 - Wait Cycle 1 Register

The WCY1 register is divided into four fields as shown in Figure 5-4. The first three fields define the number of wait states inserted into three different regions of memory defined by the WMB register. The Lower Memory Wait (LMW) field defines the number of wait states inserted for memory accesses into the lower memory region. The Middle Memory Wait (MMW) field defines the number of wait states inserted for memory accesses into the middle memory region. The Upper Memory Wait (UMW) field defines the number of wait states inserted for memory accesses into the upper memory region.

The ZT 8802 does not require any memory wait states if onboard memory devices with access times less than 200 ns are used. STD bus memory and I/O require one wait state for correct operation. STD ROM and Ziatech DOS software programs the WCY1 register with a 44h. This allows for one wait state for offboard (STD bus) memory and I/O.

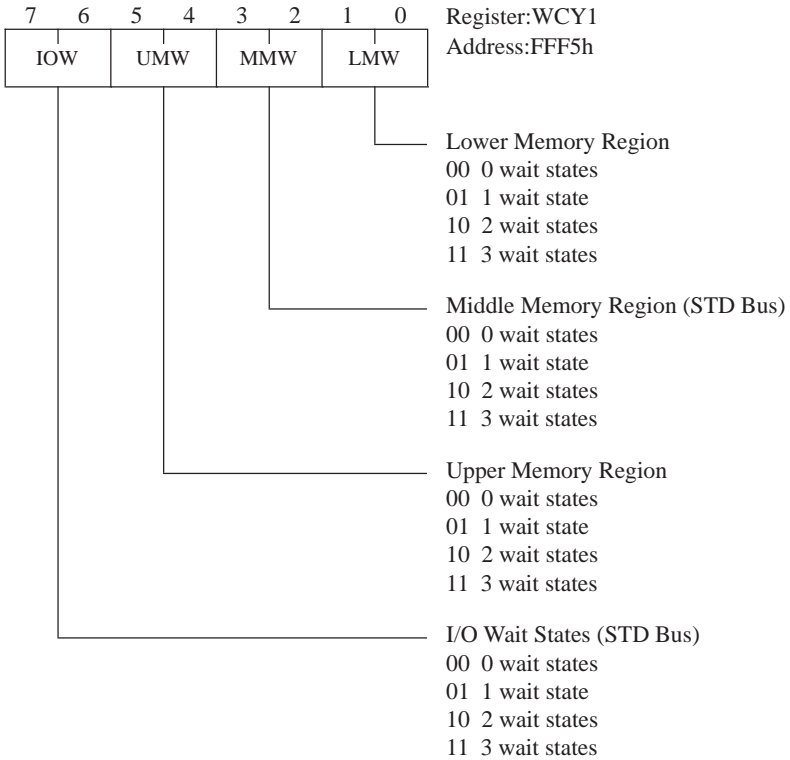
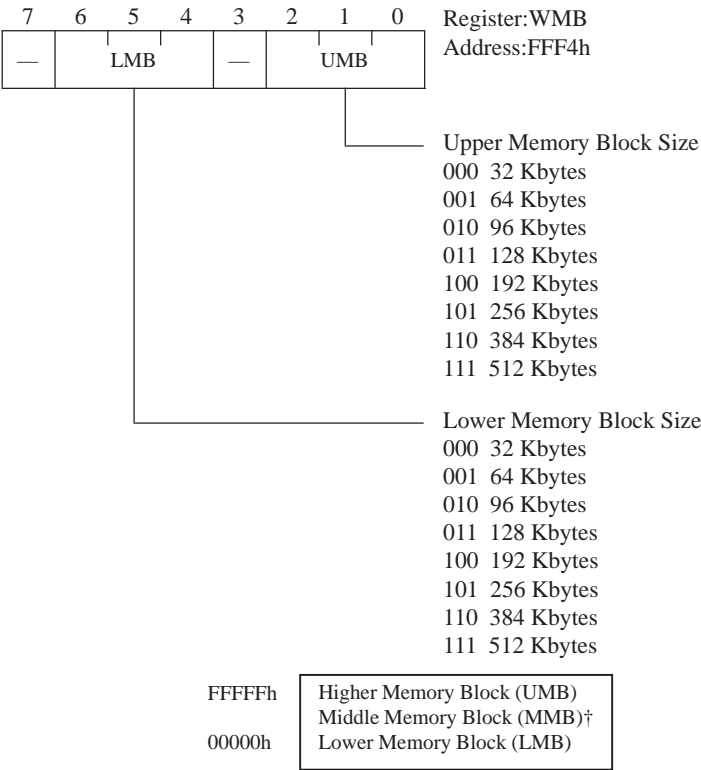


Figure 5-4. Wait-Cycle 1 Register.

WMB - Wait Memory Boundary Register

The ZT 8802 does not require any wait memory wait states if onboard memory devices with access times less than 200 ns are used. If slower memory devices are used, the WMB register divides the ZT 8802 memory into three regions and the WCY1 register defines the number of wait states inserted into each. As shown in Figure 5-5, the WMB register is divided into Upper Memory Boundary (UMB) and Lower Memory Boundary (LMB) fields.



† MMB is defined as the address range between LMB and UMB.

Figure 5–5. Wait-Cycle Memory Boundary Register.

The Middle Memory Block is defined between the top of the LMB and the bottom of the UMB. Offboard memory (STD bus) requires one wait state in all cases and is defined by the MMB.

The LMB field defines the lower memory address range starting from zero. This field can be programmed to include the memory devices inserted into the RAM LOW and RAM HIGH sockets. The UMB field defines the upper memory address range ending at FFFFh. This field can be programmed to include the memory device inserted into the ROM socket. The STD ROM software initializes all bits of the WMB register to 33h. Ziatech DOS configures this register differently for differing amounts of on-board memory. All off-board memory regions are programmed for one wait-state. All on-board regions, including the BIOS, are initialized for zero wait states.

RFC - Refresh Control Register

The ZT 8802 does not use the refresh controller. To prevent the refresh controller from affecting system performance, bit 7 must be programmed with a logical 0, as shown below.

7	6	5	4	3	2	1	0	Register:RFC
0	—	—	—	—	—	—	—	Address:FFF2h

Figure 5–6. Refresh Control Register.

TCKS - Counter/Timer Clock Selection Register

The V40 includes three programmable counter/timers. The TCKS register, shown in Figure 5-7, selects the clock source for each counter/timer and a frequency divisor used by all three counter/timers.

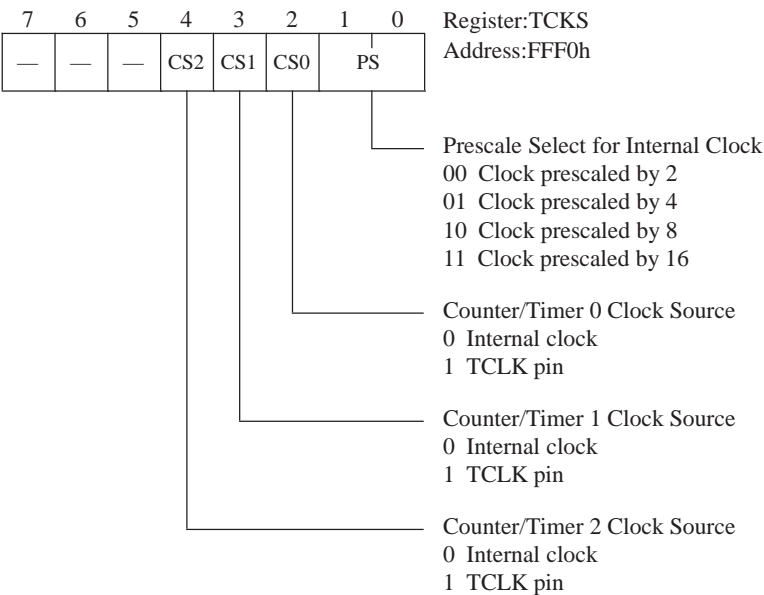


Figure 5–7. Counter/Timer Clock Selection Register.

The CS0, CS1, and CS2 (Clock Select 0, 1, and 2) bits select the counter/timer clock source to be the reference clock internal to the V40 or the TCLK pin available on an external V40 pin. The V40 clock operates at 8 MHz with a 50% duty cycle. The TCLK signal is jumper selectable between connector J2 and the onboard 1.19318 MHz oscillator. The Prescale (PS) field selects a prescale value that divides the clock frequency of all counter/timers using the V40 internal clock before applying it to the counter/timers.

The STD ROM software programs all bits of TCKS with logical 0s to select the internal clock as the source for all three timer/counters and prescale the clock frequency by 2.

Ziatech DOS configures this register to select the external clock frequency (1.19318 MHz) for all timer/counters and a prescale of 2. The default value for non-VSC DOS systems is 1Ch. If VSC is used, then timer/counter 1 is programmed for the internal clock frequency (16 MHz) for baud rate generation. For VSC DOS systems, the default programmed value for TCKS is 14h.

RESET

The V40 configuration registers are automatically initialized to a default state when power is applied to the V40 and also during reset from an external source such as a push-button reset. Table 5-3 shows the default state of the configuration registers before software initialization. Table 5-4 shows the default register values for STD ROM systems, and Table 5-5 shows the default register values for DOS systems.

Table 5-3
V40 Configuration Register Defaults after RESET.

Registers	Default Bit Values ^[1]							
	7	6	5	4	3	2	1	0
OPCN	-	-	-	-	0	0	0	0
OPSEL	-	-	-	-	0	0	0	0
OPHA	-	-	-	-	-	-	-	-
DULA	-	-	-	-	-	-	-	-
IULA	-	-	-	-	-	-	-	-
TULA	-	-	-	-	-	-	-	-
SULA	-	-	-	-	-	-	-	-
WCY2	-	-	-	-	1	1	1	1
WCY1	1	1	1	1	1	1	1	1
WMB	-	-	-	-	-	-	-	-
RFC ^[2]	-	-	-	0	1	0	0	0
TCKS	-	-	-	0	0	0	0	0

[1] Bit positions marked with "-" can default to 1 or 0.

[2] The refresh enable bit of the RFC register is not affected by resets other than power on.

Table 5-4
STD ROM V40 Configuration Register Defaults.

Registers	Default Bit Values ^[1]							
	7	6	5	4	3	2	1	0
OPCN	-	-	-	-	0	1	1	0
OPSEL	-	-	-	-	1	1	1	0
OPHA	0	0	0	0	0	0	0	0
DULA	1	1	0	1	0	0	0	0
IULA	0	0	1	0	0	0	0	0
TULA	0	1	0	0	0	0	0	0
SULA	1	0	1	1	0	0	0	0
WCY2	0	0	0	0	1	0	0	0
WCY1	0	1	0	0	0	1	0	0
WMB	0	0	1	1	0	0	1	1
RFC	0	0	0	0	0	0	0	0
TCKS	0	0	0	0	0	0	0	0

[1] Bit positions marked with "-" can default to 1 or 0.

Table 5-5
DOS System V40 Configuration Register Defaults.

Registers	Default Bit Values ^[1]							
	7	6	5	4	3	2	1	0
OPCN	-	-	-	-	0	0	1	0
OPSEL	-	-	-	-	1	1	1	1
OPHA	0	0	0	0	0	0	0	0
DULA	1	1	0	1	0	0	0	0
IULA	0	0	1	0	0	0	0	0
TULA	0	1	0	0	0	0	0	0
SULA	1	0	1	1	0	0	0	0
WCY2	0	0	0	0	0	0	0	0
WCY1	0	1	0	0	0	1	0	0
WMB ^[2]	0	1	0	1	0	0	1	1
RFC	0	0	0	0	0	0	0	0
TCKS ^[3]	0	0	0	1	1	1	0	0

[1] Bit positions marked with "-" can default to 1 or 0.

[2] This value will change depending upon the amount of onboard memory. The value shown is for a 256K system.

[3] This value is for non-VSC systems. VSC systems will be programmed to 14h.

Chapter 6

COUNTER/TIMERS (V40)

Contents	Page
OVERVIEW	6-2
ZT 8802 SPECIFICS	6-3
PROGRAMMABLE REGISTERS	6-5
Timer Mode Register (TMD)	6-6
Count Registers	6-10
Status Registers	6-10
ADDITIONAL INFORMATION	6-11

OVERVIEW

This chapter describes the Counter/Timer Control Unit (TCU) and provides register descriptions.

The TCU includes three 16-bit programmable counter/timers. Applications for these counter/timers include baud rate generation for the serial controller, timing loops, timed and periodic interrupts, and asynchronous event counters.

The main features of the TCU are:

- Three 16-bit counter/timers
- Six programmable operating modes
- Binary and BCD counting
- Interrupt and polled operation
- Functionally equivalent to 8254

ZT 8802 SPECIFICS

The clock source for each counter/timer is defined in the TCKS V40 configuration register. The choices for the clock source are the V40 internal clock and the TCLK signal. The clock internal to the V40 has a frequency of 8 MHz and a duty cycle of 50%. The TCLK signal is available through connector J2. Optionally, the TCLK signal may be driven by the on-board timer tick oscillator at a rate of 1.19318 MHz, which is consistent with the IBM PC in frequency for the counter/timers. The TCLK signal must meet the following requirements:

- Operating frequency between DC and 8 MHz
- Rise and fall times less than 25 nanoseconds
- Clock low and clock high times greater than 50 nanoseconds

Counter/Timers 0 and 1 have implied uses because of their dedicated output connections to other devices internal to the V40. The output of Counter/Timer 0 is connected to IRQ0 of the interrupt controller. This dedicates Counter/Timer 0 to generating timed or periodic interrupts.

The output of Counter/Timer 1 is connected to the V40 serial port for baud rate generation. The output of Counter/Timer 1 can also be connected to IRQ2 of the interrupt controller if the V40 serial port is not needed. This connection is made with the OPCN V40 configuration register.

Counter/Timer 2 does not have an implied use. Counter/Timer 2 is used in IBM PCs and compatibles for the speaker frequency. The generating speaker is used by DOS through a BIOS function call for reporting keyboard errors and is available for application software. Customers using Counter/Timer 2 must be aware of this, even though there is no speaker on board. A software application may set Counter/Timer 2 for a particular frequency, yet a keyboard error will cause the BIOS to come along later and change it. The output (TOUT2) and control (TCTL2) signals for Counter/Timer 2 are available through connector J2 to be used as required by the application. See Figure 6-1 for J2 configuration.

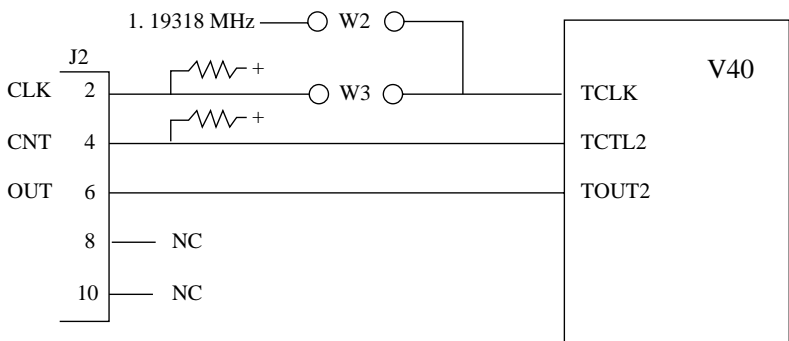


Figure 6-1. J2 Counter/Timer 2 Connector.

PROGRAMMABLE REGISTERS

Four separately addressable registers are provided for communication to the TCU. The TMD (Timer Mode) register specifies the operation of the three counter/timers. The TMD is a write-only register. The other three bidirectional registers are used to write the count to the counter/timers and read back the count and status. These registers are called the Count and Status registers.

The base I/O address of the TCU registers is defined by the OPHA and TULA registers. OPHA is programmed with the high byte and TULA with the low byte of the 16-bit address. Refer to Chapter 5 for a complete discussion on the V40 configuration registers, including OPHA and TULA. Table 6-1 shows the address of the TCU registers relative to the base address.

Table 6-1
TCU Register Addressing.

Address	Register	Operation	Page Number
Base + 0	TCT0 Count	Read/Write	6-10
Base + 0	TCT0 Status	Read	6-10
Base + 1	TCT1 Count	Read/Write	6-10
Base + 1	TCT1 Status	Read	6-10
Base + 2	TCT2 Count	Read/Write	6-10
Base + 2	TCT2 Status	Read	6-10
Base + 3	TMD	Write	6-6

Timer Mode Register (TMD)

The counter/timers must be initialized with the 8-bit TMD register. The three formats for the TMD register are shown in Figures 6-2, 6-3, and 6-4 (pages 6-7 to 6-9). The General Mode format is programmed initially to define the operation of the counter/timers. The Count Latch Mode and Multiple Latch Mode are programmed at any time to read the count and status data while the counter/timers are operating.

General Mode

The General Mode format, shown in Figure 6-2, specifies the operating mode of the individual counter/timers. The Select Counter bits specify the Multiple Latch command or which counter/timer will receive the mode. Selecting the Multiple Latch command changes the definition of the TMD register bits to that of the Multiple Latch Mode format. The following bit definitions apply only if the Multiple Latch and Count Latch options are not programmed.

The Read/Write Mode bits specify the Count Latch command or the format of the count transferred between the CPU and the TCU. Selecting the Count Latch command redefines the bits of the TMD register to that of the Count Latch Mode format.

The count transferred to or from the 16-bit counter/timers is one or two 8-bit values depending on the Read/Write Mode bits. If the low byte option is chosen, the 8-bit count transferred to the counter/timer is placed into the low byte of the Down Counter and the high byte is automatically set to 0. The high byte option means the 8-bit count is transferred to the upper byte of the Down Counter with the low byte set to 0. Selecting the two-byte option prepares the counter/timer to receive two bytes, placing the first into the lower byte of the Down Counter and the second into the upper.

A new count can be written into the counter/timers at any time without reprogramming the TMD register. Care must be taken to be consistent with the Read/Write Mode each time the new count is programmed. As an example, assume that Counter/Timer 0 is programmed with a Read/Write Mode of two bytes. Two bytes must be written to Counter/Timer 0 each time a new count is specified. The same applies for reading Counter/Timer 0; that is, two count bytes must be read at a time.

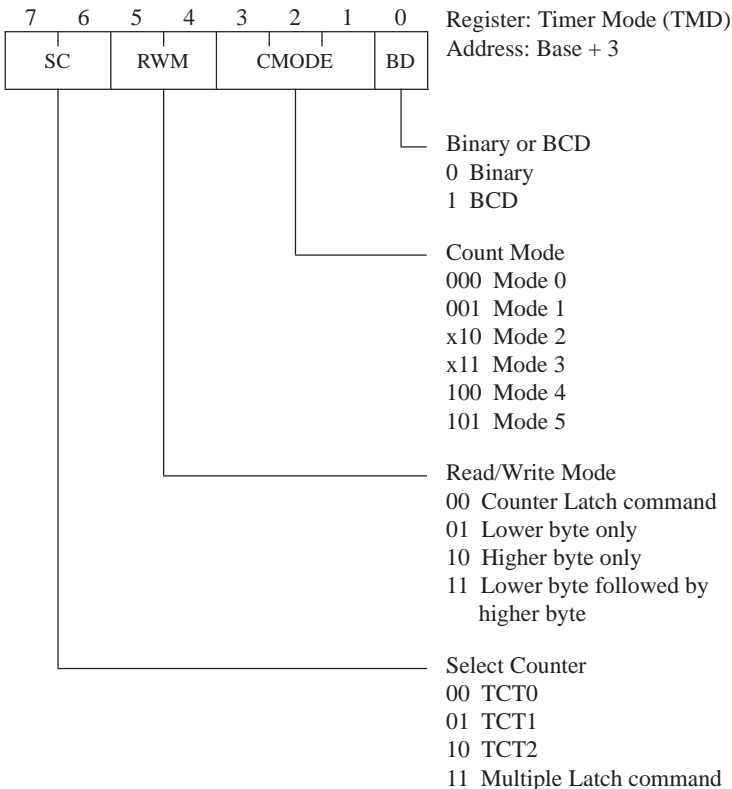


Figure 6–2. Counter/Timer General Mode Register.

Each counter/timer must be programmed to operate in one of the six possible count modes. The selection of the count mode is based on the needs of the application. The counting operation of each counter/timer is programmed as binary or Binary-Coded-Decimal (BCD). The range of a counter/timer programmed for binary operation is 0 to FFFFh, while BCD operation is 0 to decimal 9999.

Count Latch Mode

The Count Latch Mode, shown in Figure 6-3, requires that the first six bits be set to a logical 0. The Select Counter bits specify which counter/timer's data is to be latched.

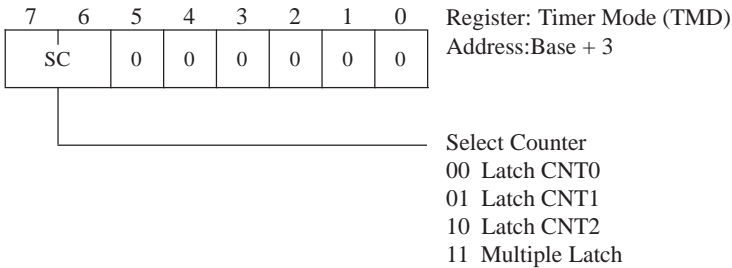


Figure 6-3. Counter/Timer Count Mode Register.

Multiple Latch Mode

Programming the Select Counter bits of the TMD to logical 1s defines the Multiple Latch command. The format of the Multiple Latch Mode is shown in Figure 6-4. The CNT0, CNT1, and CNT2 bits select which of the counter/timers is latched. The Status Latch and Count Latch bits determine if the status or count, or both, should be latched. The status must be latched to be read. The count can be read without being latched, but it is invalid if it is changing at the time of the read.

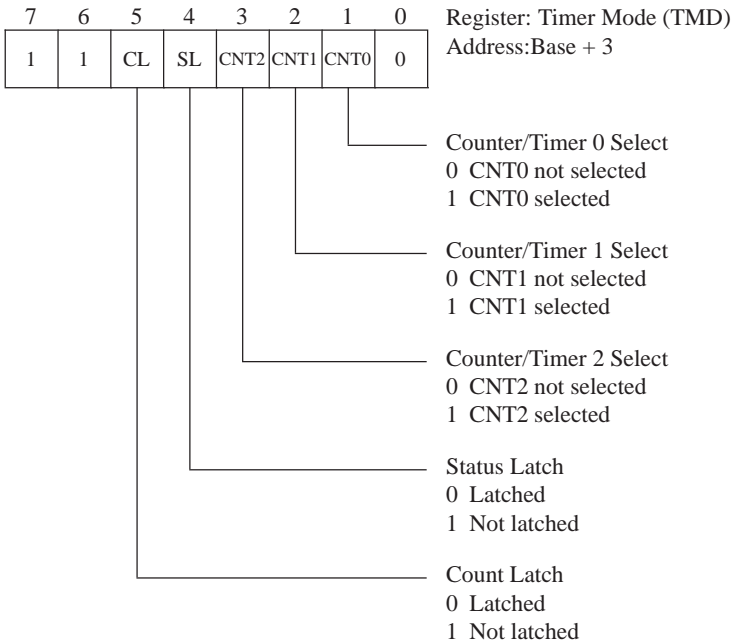


Figure 6-4. Counter/Timer Multiple Mode Register.

Count Registers

The Count register is illustrated in Figure 6-5. Unlike the Mode register, there is one Count register for each of the three counter/timers. The Count register transfers count values to and from the Down Counter. The 16-bit register is programmed with a high byte, low byte, or both high and low byte as specified with the Read/Write Mode bits in the Mode register. If the high byte or low byte mode is selected, only one read or write operation is needed for data transfers. Two read or write operations are required for the two-byte mode, with the low byte transferred first, then the high byte.

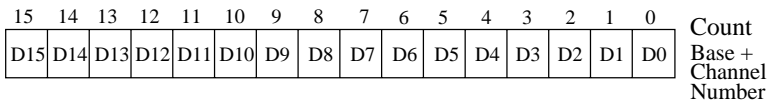


Figure 6-5. Counter/Timer Count Register.

Status Registers

Each counter/timer includes a Status register. The status can be read at any time from the Status register. The format for the status is shown in Figure 6-6.

The first six bits of the Status register provide information about the programmed state of the selected counter/timer. This information is in the Status register to prevent the application software from having to save it. The Null Count bit flags when the last count written to the Count register is transferred to the Down Counter. This is designed to prevent the application software from reading the Down Counter before it is updated to the last count written. The Output Level bit contains the current state of the counter/timer output (TOUT).

You must use the Multiple Latch command to read the status. The number of required read operations depends on the Read/Write Mode and the Multiple Latch command. If the Read/Write Mode is high byte or low byte and only the status is latched with the Multiple Latch command, one read operation is all that is needed. Two reads are

required if both the status and the count are latched by the Multiple Latch command. The first read is for the status and the second is for the data. If the Read/Write Mode is programmed for two byte transfers and the Multiple Latch command is programmed to latch only the status, one read is all that is required. If both the status and data are latched, three reads are required. The first read is for the status and the next two are for the low and high byte of the count, respectively.

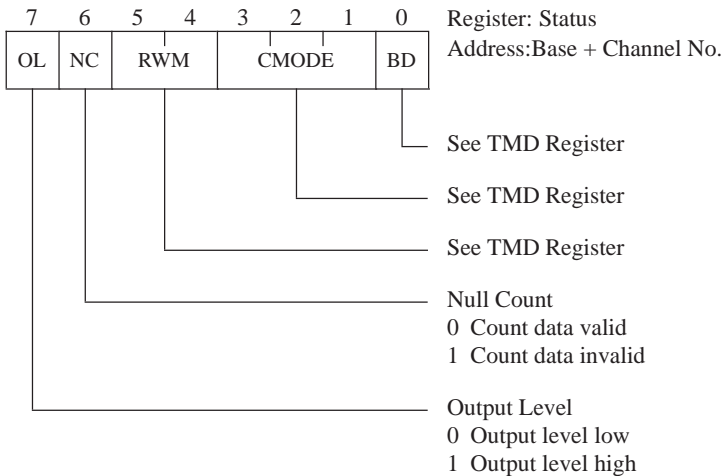


Figure 6–6. Counter/Timer Status Register.

ADDITIONAL INFORMATION

Refer to the NEC *16-Bit V Series Microprocessor* data book for more information on the V40 counter/timers.

Chapter 7

INTERRUPT CONTROLLER (V40)

Contents	Page
OVERVIEW	7-2
ZT 8802 SPECIFICS	7-3
Cascade Mode	7-5
PROGRAMMABLE REGISTERS	7-6
Initialization Words (IIW1, IIW2, IIW3, and IIW4)	7-7
Operation Words (IMKW, IPFW, and IMDW)	7-11
Status Words (IRQ, IIS, and IPOL)	7-15
ADDITIONAL INFORMATION	7-18

OVERVIEW

This chapter describes the Interrupt Control Unit (ICU) and provides register descriptions.

The ICU is a programmable interface between interrupt generating peripherals and the CPU. The ICU monitors eight interrupt inputs with programmable priority. When peripherals request service, the ICU interrupts the CPU with a pointer to a service routine for the highest priority device. This type of interrupt management is needed for an efficient interface between the CPU and supporting peripheral devices, such as serial controllers and counter/timers. The major features of the ICU are as follows:

- Eight individually maskable interrupts
- Level-triggered or edge-triggered interrupts
- Fixed and rotating prioritization
- Status available for polled operation
- Functionally equivalent to the 8259

ZT 8802 SPECIFICS

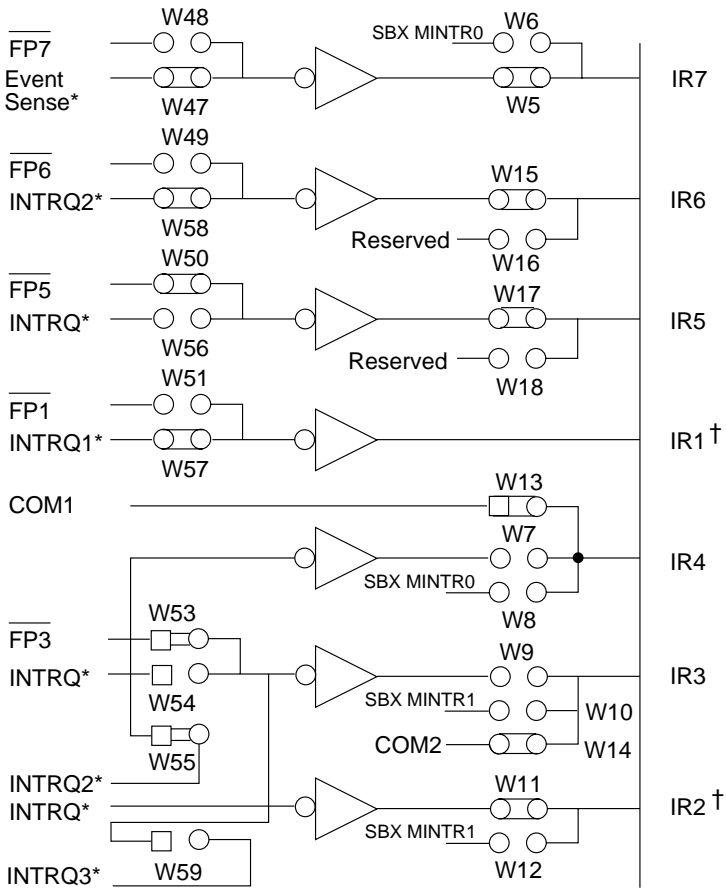
Possible inputs to the interrupt controller are connected as shown in Table 7-1.

Table 7-1
Interrupt Controller Inputs.

Input	Connection
IRQ0	Counter/Timer 0 output
IRQ1	V40 Serial or $\overline{FP1}$ or [INTRQ1*]
IRQ2	Counter/Timer 1 or [INTRQ*] or SBX Expansion Module Interrupt 1
IRQ3	[COM2 Interrupt] or INTRQ* or INTRQ2* or $\overline{FP3}$ or SBX Expansion Module Interrupt 1 or INTRQ3*
IRQ4	[COM1 Interrupt] or INTRQ* or INTRQ2* or SBX Expansion Module Interrupt 0
IRQ5	INTRQ* or $\overline{FP5}$
IRQ6	[INTRQ2*] or $\overline{FP6}$
IRQ7	[EVENT SENSE*] or $\overline{FP7}$ or SBX Expansion Module Interrupt 0

Square brackets [] denote default.

Figure 7-1 below shows the possible interrupt sources for the interrupt input requests IR1-IR7. The IR4 and IR3 levels may be alternately jumpered by using right angle jumpers at W54 and W55, as shown in Figure 7-2.



† IR1 and IR2 have other options within the V40. Refer to the OPCN register in Chapter 5 for details.

Figure 7-1. Interrupt Jumper Selection.

Cascade Mode

The interrupt controller includes a cascade mode supported by the ZT 8802. Cascade mode is a scheme by which to expand the number of interrupt request inputs by connecting up to seven slave interrupt controllers to the inputs of the master interrupt controller. In this scheme, the master interrupt controller must provide a cascade address to the slave controllers during interrupt acknowledge. The cascade address is provided by the ZT 8802 for interrupt levels that are programmed in cascade mode.

When the master interrupt controller is programmed for cascade mode on a particular interrupt level, an interrupt on that level causes the master interrupt controller to initiate a cascade interrupt sequence. The master interrupt controller supplies a 3-bit address (on A8-10 of the backplane) while driving the INTAK* signal on the backplane. The slave interrupt controller responds by driving a vector onto the data lines during the second INTAK* pulse. The host CPU (ZT 8802) then uses this vector to index a table. This table supplies the address in memory of the interrupt service routine. Up to eight interrupts on the slave interrupt controller can then be supported by one master interrupt controller input.

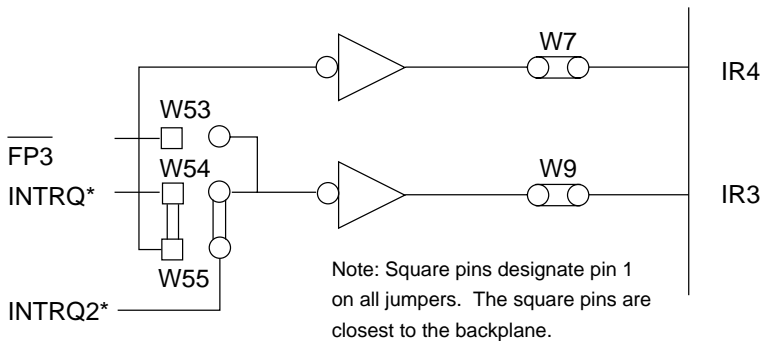


Figure 7-2. Alternate IR4/IR3 Jumper Selections.

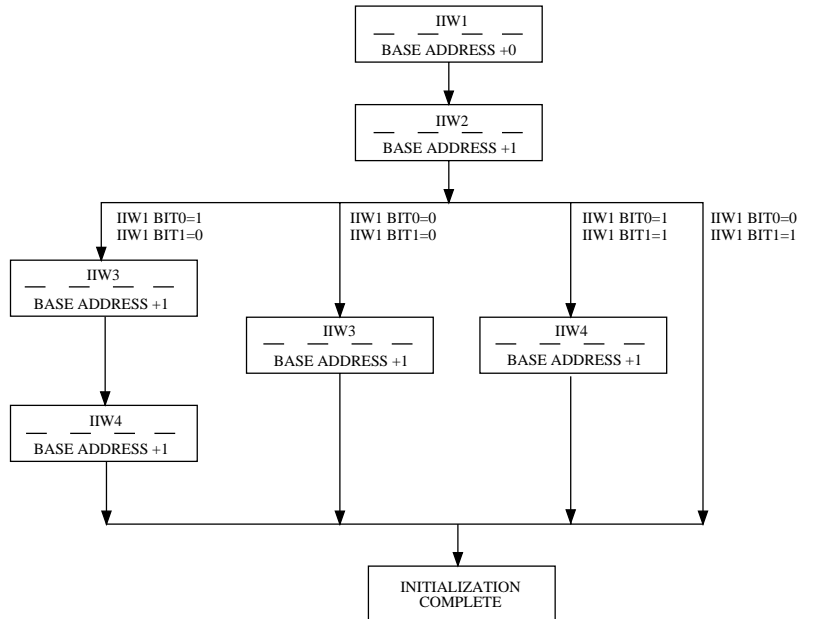
PROGRAMMABLE REGISTERS

The ICU is initialized with Interrupt Initialization Word 1 (IIW1) through Interrupt Initialization Word 4 (IIW4). Once initialized, the operation of the ICU is controlled with the Interrupt Mask Word (IMKW), Interrupt Priority and Finish Word (IPFW), and Interrupt Mode Word (IMDW). There are also three status words that can be read to interrogate the operation of the ICU: the Interrupt Request (IRQ), Interrupt In-Service (IIS), and Interrupt Poll (IPOL). Please note that the "word" reference does not mean the values are 16 bits; all communication to the ICU is done through eight-bit data.

All initialization, operation, and status words are accessed through two I/O addresses as shown in Table 7-2. (The base address is selected with the OPHA and IULA V40 configuration registers; see page 5-6 for details.) As might be expected, a specific sequence of read and write operations is needed to pass multiple bytes through a single I/O address. A complete description of all the programmable words and how they are accessed is given on the following pages.

Table 7-2
ICU Register Addressing.

Address	Value	Operation	Page No.
Base + 0	IRQ, IIS, IPOL	Read	7-15
Base + 0	IIW1	Write	7-7
Base + 0	IPFW, IMDW	Write	7-12, 7-14
Base + 1	IMKW	Read/Write	7-11
Base + 1	IIW2, IIW3, IIW4	Write	7-7



IIW1 and IIW2

Interrupt Initialization Words 1 (IIW1) and 2 (IIW2) are required for ICU initialization. The IIW1 register, shown in Figure 7-4, is divided into three fields labeled II4 (Interrupt Initialization 4), II3 (Interrupt Initialization 3), and LEV (Level). The II4 bit selects whether or not the IIW4 register is to be programmed. If II4 is set to a logical 1, the ICU expects IIW4 to be written as part of the initialization. The II3 bit selects whether or not the IIW3 register is to be programmed. If II3 is set to a logical 0, the ICU expects IIW3 to be written as part of the initialization. The LEV bit of IIW1 selects between level- or edge-triggered interrupt request inputs. Setting LEV to a logical 1 selects level-triggered inputs, while a logical 0 selects edge-triggered.

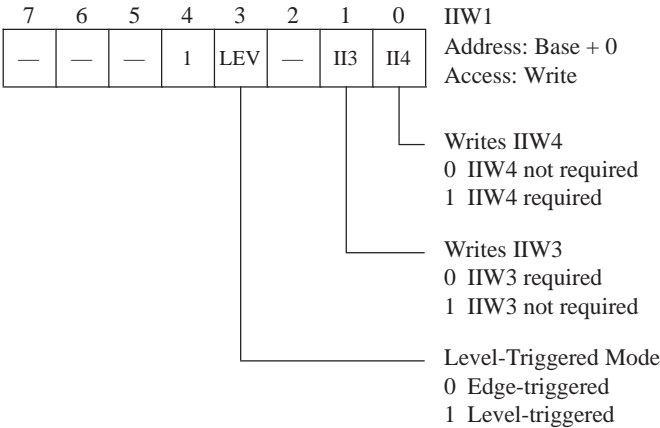


Figure 7-4. Interrupt Initialization Word 1.

The ICU responds to an interrupt acknowledge by supplying the CPU with an interrupt vector based on which interrupt generated the request and the value programmed into IIW2. The format for IIW2 is shown in Figure 7-5. Bits V3 through V7 define the upper five bits of the vector address.

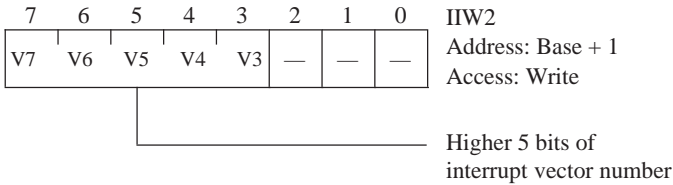


Figure 7-5. Interrupt Initialization Word 2.

IIW3

The ZT 8802 does support cascading the interrupt controller inputs to other interrupt controllers. Bits 0 through 7 must be programmed with logical 0s, as shown in Figure 7-6, for the levels that are not to be used in cascade mode. Program a 1 for all levels that are to be used in cascade mode. IIW1 must be programmed with a logical 0 in the II3 bit if IIW3 is used. Common bus practice for cascaded interrupts is to drive the INTRQ* signal from the slave PIC, which is normally jumpered to drive interrupt level 2, on the master PIC.

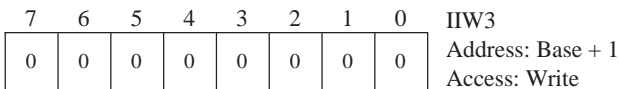


Figure 7-6. Interrupt Initialization Word 3.

IIW4

Figure 7-7 shows the architecture for IIW4. IIW1 must be programmed with a logical 1 in the II4 bit if IIW4 is used. A logical 1 in the SFI bit enables the Self Finish Interrupt and a logical 0 disables it. The interrupt service routine must include an EOI command when the Self Finish Interrupt is disabled.

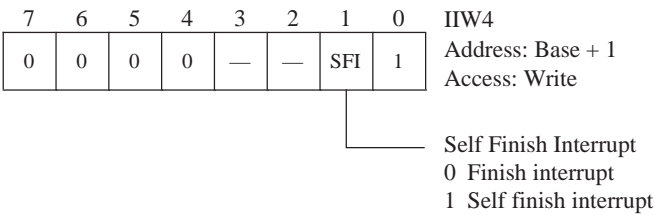


Figure 7–7. Interrupt Initialization Word 4.

Operation Words (IMKW, IPFW, and IMDW)

Once initialized, the operation of the ICU is controlled with three 8-bit values called the Interrupt Mask Word (IMKW), Interrupt Priority and Finish Word (IPFW), and the Interrupt Mode Word (IMDW). The Operation Words can be transferred in any sequence to perform such functions as enabling and disabling individual interrupt requests and changing interrupt priorities.

IMKW

The IMKW masks interrupt request inputs. Interrupts are masked by writing IMKW to the IMK register. IMKW can be read directly from the IMK register to determine the current status of the mask. This eliminates the need for application software to maintain a copy of the mask in program memory.

As shown in Figure 7-8, each of the eight bits in IMKW represents an interrupt input. Bit M0 is used to mask IRQ0, M1 is used to mask IRQ1, and so on. Setting a bit in IMKW to a logical 1 prevents the interrupt request for the respective input from being acknowledged by the ICU. The interrupt request is latched in the IRQ register but it never reaches the IIS register.

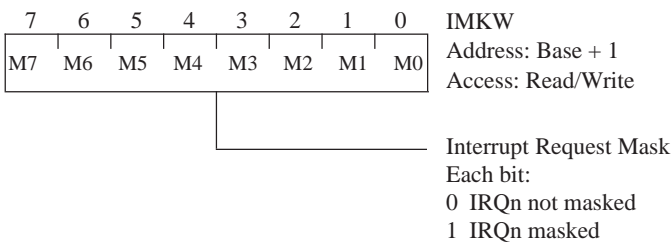


Figure 7-8. Interrupt Mask Word.

IPFW

IPFW selects fixed or rotating priorities and the method of informing the ICU that an interrupt has been serviced. Operation of the ICU can be changed at any time by writing a new IPFW. Refer to Figure 7-9 when programming the IPFW.

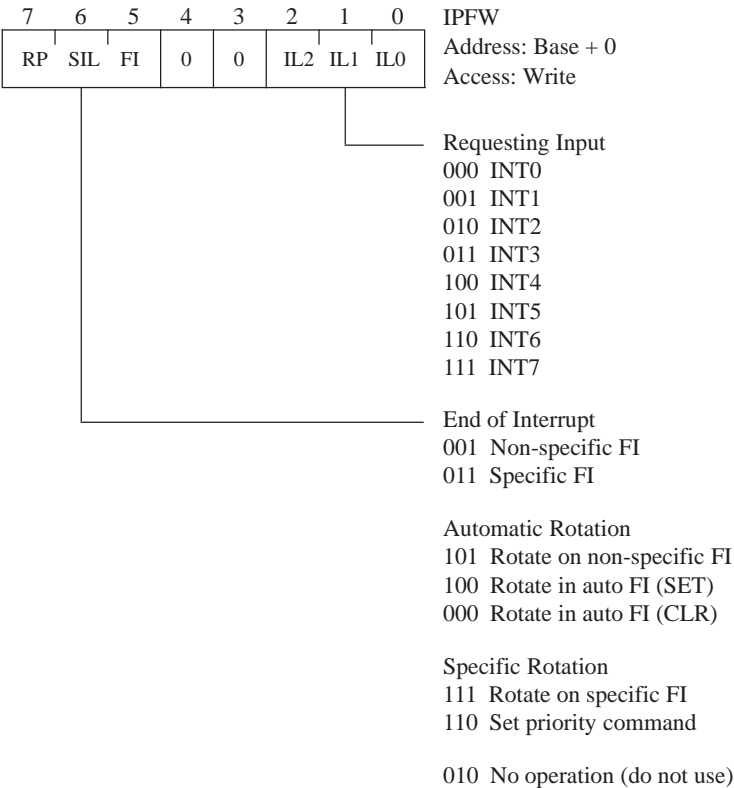


Figure 7-9. Interrupt Priority and Finish Word.

The IL0 through IL2 bits designate an interrupt level. This level is used by certain combinations of the FI, SIL, and RP bits to either reset an interrupt request that has been recognized or set a specific priority.

The ICU uses the IIS register to keep track of which interrupts are being serviced and their relative priorities. The ICU updates the IIS register based on a Finish Interrupt command. There are three methods to generate the finish interrupt command: specific FI, non-specific FI, and automatic FI. The automatic FI is programmed with IIW4. The specific and non-specific FI are selected with the FI bit. A logical 1 in FI enables the specific or non-specific FI based on the SIL and RP bits.

The SIL bit enables bits IL0 through IL2 for selected operations. IL0 through IL2 indicate an interrupt level to be reset during the finish interrupt commands or a new priority for priority rotation commands.

The ICU provides several methods by which to establish priorities for the interrupt request inputs. The RP bit selects the priority rotation options. A logical 1 in the RP bit indicates that rotation in priorities is to take place based on the values of the FI and SIL bits. A logical 0 in the RP bit means no priority rotation will take place.

IMDW

IMDW controls the method of reading status from the ICU and enables a special type of interrupt masking.

The format of the IMDW is shown in Figure 7-10. The first two bits are used to select the IRQ and IIS registers so they can be read by the application software. A logical 1 in bit 0 selects the IIS register and a logical 0 selects the IRQ register. A logical 1 in the SR bit (Select Register, bit 1) enables the reading of the IRQ and IIS registers.

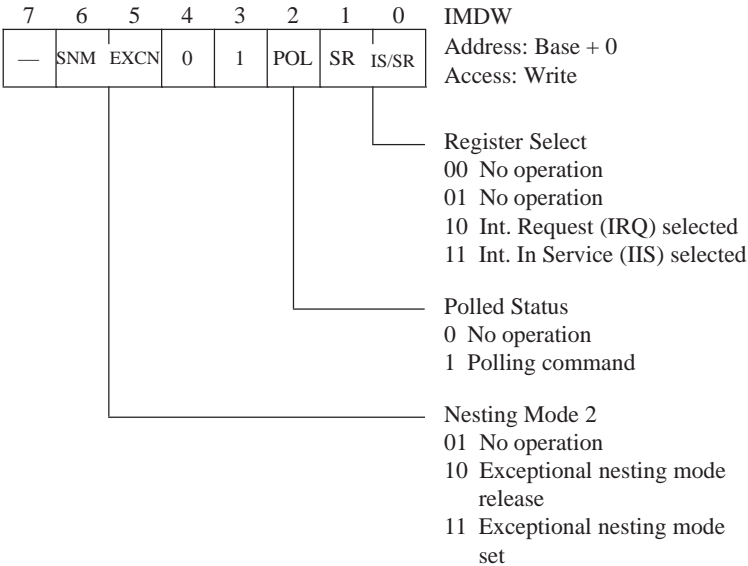


Figure 7–10. Interrupt Mode Word.

The POL bit selects the poll command. The two most commonly used methods to service peripherals in a microprocessor system are polling and interrupts. Although interrupts are the fastest method to service peripherals, using the ICU in a polled operation is still faster than polling each peripheral one at a time. Setting the POL bit to a logical 1 enables the reading of the poll status. The POL bit overrides the SR bit if they are both set.

The EXCN and SNM bits can be programmed to enable or disable the exceptional nesting mode of operation. The exceptional nesting mode is armed by setting SNM (Set Nesting Mode) to a logical 1. The EXCN bit can then be used to set or release the exceptional nesting mode. This operating mode is used to permit interrupts of lower priority than the one currently under service to be recognized.

Status Words (IRQ, IIS, and IPOL)

Three 8-bit status words can be read from the ICU. These are the Interrupt Request (IRQ), Interrupt In-Service (IIS), and Interrupt Poll (IPOL). These words can be read at any time by programming the first three bits of the IMDW. Once the IMDW is programmed to select one of the status words, that word can be read as many times as needed. The IMDW must be programmed with a new value to read another of the status words. The formats of the IRQ, IIS, and IPOL are illustrated on the following pages.

IRQ and IIS

The IRQ and IIS status words, shown in Figure 7-11 below, are taken directly from the Interrupt Request register and Interrupt In-Service register, respectively. The IRQ status word contains all the interrupt levels requesting service. The IIS status word contains all the interrupt levels currently being serviced. Bit 0 of both status words corresponds to IRQ0; bit 1 corresponds to IRQ1, and so on.

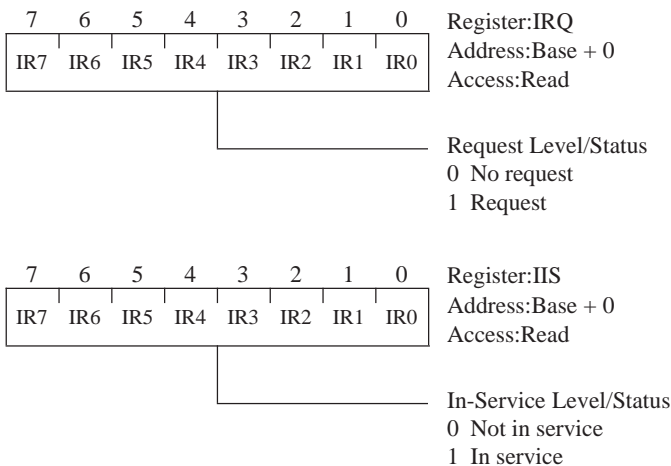


Figure 7-11. Interrupt Status Registers IRQ and IIS.

IPOL

In most applications, polling and interrupts are generally used to service peripherals. The ICU can be used in a polled system to increase the efficiency of servicing peripherals, even though the ICU is designed primarily for interrupt control. The efficiency is increased because the CPU can poll the ICU to determine the status of several peripheral devices at one time.

Figure 7-12 shows the IPOL status word. Bits PL0 through PL2 define the highest priority interrupt input requesting service. For example, if all three bits are set to a logical 1, then IRQ7 is the highest priority request.

The INT bit indicates whether there are any interrupt requests. A logical 1 signals an interrupt request and a logical 0 signals no interrupt request. If INT is a logical 0, PL0 through PL2 are all set to logical 1. The typical polling sequence is to set the POL bit in IMDW, read the IPOL register, and test the INT bit. If INT is a logical 1, decode PL0 through PL2 to determine which peripheral to service.

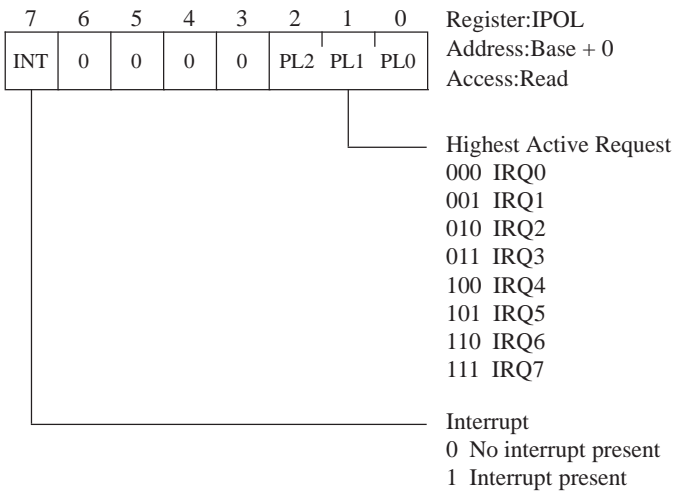


Figure 7–12. Interrupt Status Register IPOL.

ADDITIONAL INFORMATION

Refer to the NEC *16-Bit V Series Microprocessor* data book for more information on the V40 interrupt controller.

Chapter 8

DMA CONTROLLER (V40)

Contents	Page
OVERVIEW	8-2
ZT 8802 SPECIFICS	8-3
Bus Request/Bus Acknowledge	8-3
PROGRAMMABLE REGISTERS	8-4
DMA Initialize Command (DICM)	8-5
DMA Channel (DCH)	8-5
DMA Base Count/Current Count (DBC/DCC)	8-7
DMA Base Address/Current Addr. (DBA/DCA)	8-8
DMA Device Control (DDC)	8-9
DMA Mode (DMD)	8-10
DMA Status (DST)	8-11
DMA Mask (DMK)	8-12
ADDITIONAL INFORMATION	8-12

OVERVIEW

This chapter describes the Direct Memory Access Control Unit (DCU) and provides register descriptions.

The DCU is a programmable peripheral device normally used to direct high speed data transfers between memory and I/O. The DCU may also be used in a cascade mode to allow external bus masters access to the system RAM on the ZT 8802. The ZT 8950 Direct Memory Access (DMA) Controller Board is utilized in this fashion.

ZT 8802 SPECIFICS

The ZT 8802 uses one of the four DMA controllers contained in the V40. DMA channel 0 is used in cascade mode to supply the handshake for external STD bus masters. The ZT 8802 supports the BUSRQ* (Bus Request)/BUSAK* (Bus Acknowledge) protocol for transferring ownership of bus resources to an external board.

Bus Request/Bus Acknowledge

When channel 0 is programmed in cascade mode, the ZT 8802 supports external bus masters in the STD bus. When a bus master drives the BUSRQ* signal on the backplane, the DCU issues a hold request to the CPU. Upon completion of the current instruction cycle, the CPU releases control to the DCU. The DCU then drives the BUSAK* signal on the backplane, indicating to the STD bus master that it now can perform bus cycles. In this mode, the ZT 8802 allows the bus master access to all on-board memory. When BUSRQ* is released, the ZT 8802 releases BUSAK* and control reverts to the CPU.

The ZT 8950 DMA controller is an example of an STD bus master that uses this protocol. The ZT 8950 also supplies floppy disk capability for DOS systems. Ziatech DOS programs DCU channel 0 for cascade mode to allow the ZT 8950 access to system memory. The DMA controller on the ZT 8950 transfers data between the floppy disk and system memory after gaining control of the bus via the above BUSRQ*/BUSAK* sequence.

PROGRAMMABLE REGISTERS

The DCU occupies 16 consecutive I/O port addresses. Of those 16 addresses, 12 are used to access DCU functions and 4 are reserved. Table 8-1 lists the address of each of the registers relative to a programmable base address. The base address is selected with the OPHA and DULA V40 configuration registers; see page 5-6 for details.

Table 8-1
DCU Register Addressing.

Address	Register	Operation	Page Number
Base + 0	DICM	Write	8-5
Base + 1	DCH	Read/Write	8-5
Base + 2	DBC/DCC-low	Read/Write	8-7
Base + 3	DBC/DCC-high	Read/Write	8-7
Base + 4	DBA/DCA-low	Read/Write	8-8
Base + 5	DBA/DCA-middle	Read/Write	8-8
Base + 6	DBA/DCA-high	Read/Write	8-8
Base + 7	Reserved	--	--
Base + 8	DDC-low	Read/Write	8-9
Base + 9	DDC-high	Read/Write	8-9
Base + A	DMD	Read/Write	8-10
Base + B	DST	Read	8-11
Base + C	Reserved	--	--
Base + D	Reserved	--	--
Base + E	Reserved	--	--
Base + F	DMK	Read/Write	8-12

DMA Initialize Command (DICM)

The initialization command, shown in Figure 8-1, includes one bit that can be set to a logical 1 to reset the DCU. This register must be written to with the byte output instruction.

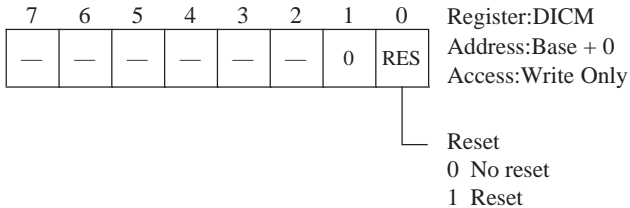


Figure 8-1. DMA Initialization Command Register.

DMA Channel (DCH)

The DMA Channel register is shown in Figure 8-2 on page 8-6. The DCH register must be accessed with byte output and input instructions. The DCH register has a different format for read and write operations. For the write operation, the BASE bit is used to select between the base and current register groups for both the address and count. Both base and current registers are written or only the current register is read if BASE is first programmed with a logical 0. Programming BASE with a logical 1 selects the base to be read or written to.

DMA Controller (V40)

For read operations, the BASE bit set to a logical 0 defines whether the current register is made available for a read operation or whether the base and current registers are written to during a write operation. A logical 1 in the BASE bit means the base register is selected.

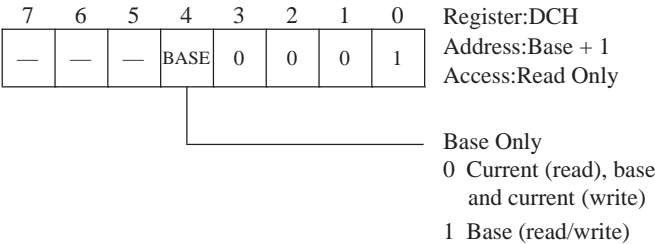
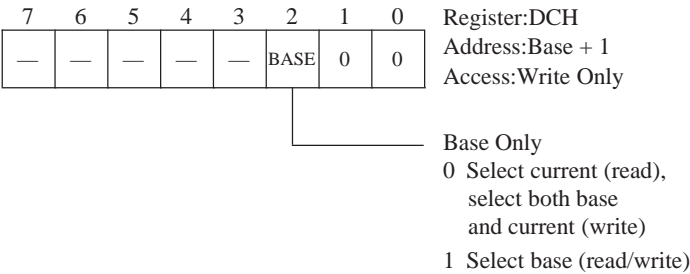


Figure 8–2. DMA Channel Register.

DMA Base Count/Current Count (DBC/DCC)

Two DBC/DCC registers make up the 16-bit DMA count as shown in Figure 8-3. The two DBC/DCC registers can be accessed with byte or word instructions. The function of these registers depends on the BASE bit of the DCH register. If the BASE bit is set to a logical 0, the values written to the Count registers are programmed into both base and current count values. The current count is read from the Count registers if the BASE bit is set to a logical 0. If the BASE bit is set to a logical 1, the values written to the Count registers are programmed into the base value only. The base value is read from the Count registers if the BASE bit is set to a logical 1. In cascade mode, these registers should be programmed to 0.

7	6	5	4	3	2	1	0	Register: DBC/DCC - Low
C7	C6	C5	C4	C3	C2	C1	C0	Address: Base + 2
								Access: Read or Write

7	6	5	4	3	2	1	0	Register: DBC/DCC - High
C15	C14	C13	C12	C11	C10	C9	C8	Address: Base + 3
								Access: Read or Write

Figure 8-3. DMA Base & Current Count Registers.

DMA Base Address/Current Addr. (DBA/DCA)

Three DBA/DCA registers specify the 20-bit address. The format of these registers is shown in Figure 8-4. The lower 16 bits of the address can be accessed with byte or word instructions. The upper four bits must be accessed with byte instructions. As is the case with the DBC/DCC registers, the BASE bit of the DCH register defines the operation of the DBA/DCA registers. A logical 0 in the BASE bit specifies that values written to the Address registers are programmed to both base and current values and data read will be current values. If the BASE bit is a logical 1, the values written to the Address registers are programmed into the base value only. The base value is read from the Count registers if the BASE bit is set to a logical 1. In cascade mode, these registers should be initialized to 0.

7	6	5	4	3	2	1	0	Register:DBA/DCA - Low
A7	A6	A5	A4	A3	A2	A1	A0	Address:Base + 4
								Access:Read or Write

7	6	5	4	3	2	1	0	Register:DBA/DCA - Middle
A15	A14	A13	A12	A11	A10	A9	A8	Address:Base + 5
								Access:Read or Write

7	6	5	4	3	2	1	0	Register:DBA/DCA - High
—	—	—	—	A19	A18	A17	A16	Address:Base + 6
								Access:Read or Write

Figure 8-4. DMA Base & Current Address Registers.

DMA Device Control (DDC)

Two DDC registers select various DCU operating modes. The format for these registers is shown in Figure 8-5. These registers can be accessed with byte or word operations. The DDMA bit can be set to a logical 1 to prevent the DCU from requesting bus access. This should be done when programming any of the DCU registers to prevent incorrect DMA operation.

The WEV bit enables or disables wait states to be inserted by the V40 WCU during the verify operation. Programming WEV with a logical 0 disables wait state insertion and programming a logical 1 enables it. The WEV bit has no effect in cascade mode.

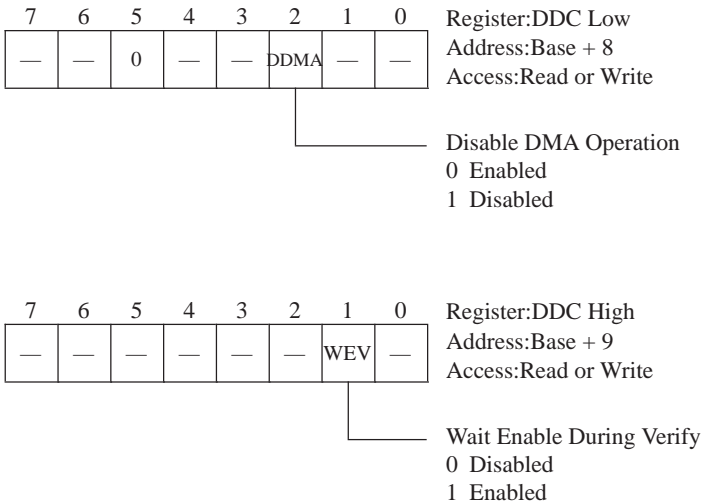


Figure 8–5. DMA Device Control Registers.

DMA Mode (DMD)

Figure 8-6 shows the format of the DMD register. The DMD register can be accessed with byte or word instructions. The TDIR field defines the mode of data transfer. A logical 0 in both bits selects the verify operation. A logical 1 in bit 2 and a logical 0 in bit 3 selects I/O to memory transfers. For memory to I/O transfers, bit 2 must be programmed with a logical 0 and bit 3 with a logical 1. The ZT 8802 supports only cascade mode operation for channel 0. For cascade mode on channel 0, program this register to C4h.

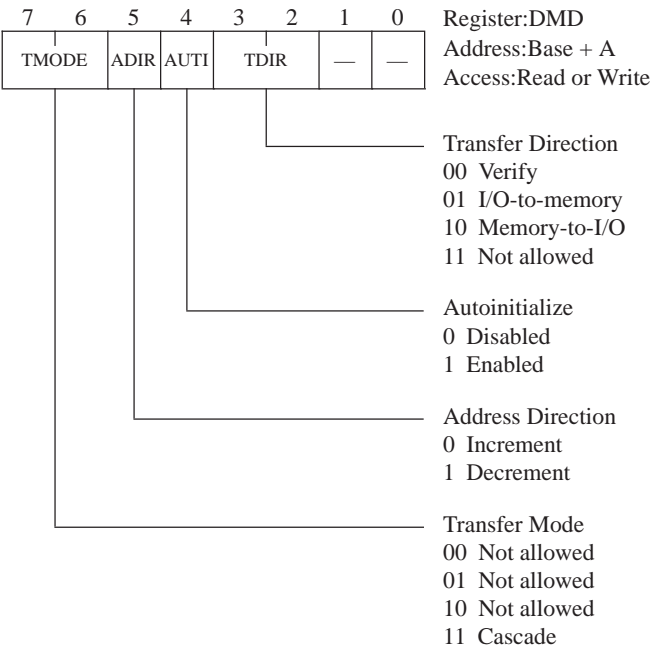


Figure 8–6. DMA Mode Register.

Autoinitialize is a feature that automatically reloads the DCU current address and count registers from the base address and count registers, respectively. The reload is done when the count register reaches zero. The autoinitialize feature is disabled by programming AUTI with a logical 0 and enabled with a logical 1.

The ADIR bit defines the operation of the DCU address adjuster. If ADIR is programmed with a logical 0, the address adjuster increments the memory address after each data transfer. If ADIR is programmed with a logical 1, the address is decremented. TMODE defines the transfer mode to be demand, single, or block.

DMA Status (DST)

The Status register includes information about the currently programmed state of the DMA channel. The format for DST is shown in Figure 8-7. DST is accessed with the byte read instruction. The TC0 bit indicates when the count register has reached zero and the DMA transfer is completed. A logical 0 in the bit position means the operation has not been terminated and a logical 1 means it has. The RQ0 bit defines the state of the DMA request input. A logical 0 indicates no request active and a logical 1 indicates a request is pending.

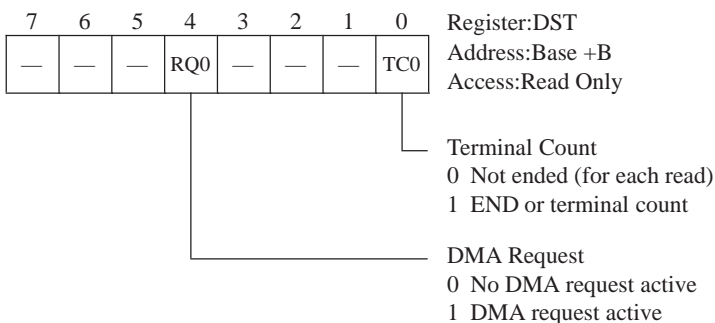


Figure 8–7. DMA Status Register.

DMA Mask (DMK)

The DMK register, shown in Figure 8-8, is used to mask DMA requests made by the DMA channel. The register is accessed with either byte write or read instructions. To mask a DMA channel, the respective bit must be programmed with a logical 1. A logical 0 enables the DMA channel to make requests. Program bit 0 to a logical 0 to enable channel 0 in cascade mode.

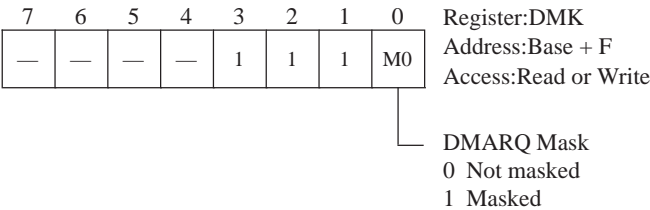


Figure 8–8. DMA Mask Register.

ADDITIONAL INFORMATION

Refer to the NEC *16-Bit V Series Microprocessor* data book for more information on the V40 DMA controller.

Chapter 9

SERIAL COMMUNICATIONS (V40)

Contents	Page
OVERVIEW	9-2
ZT 8802 SPECIFICS	9-3
RS-232-C Operation	9-3
PROGRAMMABLE REGISTERS	9-4
Serial Status Register (SST)	9-5
Serial Command Register (SCM)	9-6
Serial Mode Register (SMD)	9-7
Serial Interrupt Mask Register (SIMK)	9-8
BAUD RATE	9-9
ADDITIONAL INFORMATION	9-11

OVERVIEW

This chapter describes the Serial Control Unit (SCU) and provides register descriptions and baud rate information.

The SCU is a single serial channel that performs asynchronous serial communication between the V40 and a serial device external to the ZT 8802. This serial port is not PC (COM) compatible, but is supported by Ziatech's STD Device Driver Package (DDP), VSC under DOS, or VTI under STD ROM. Applications requiring COM port compatibility must use one of the two COM compatible ports on the ZT 8802, or an external COM port such as is available on the ZT 88CT75.

The major features of the SCU are listed below.

- Full-duplex asynchronous operation
- Clock divisor of 16 or 64
- Baud rates to 19.2 Kbaud
- Programmable character format
- Automatic break detect and handling
- Parity, overrun, and framing detection
- Interrupt and polled operation

ZT 8802 SPECIFICS

RS-232-C Operation

The SCU is implemented as a 3-wire RS-232-C serial port (see Figure 9-1). In RS-232-C mode Transmit Data (TxD), Receive Data (RxD), and Ground (GND) are used. The RS-232-C driver interface allows 5 V-only operation by using a charge pump to create RS-232-C compatible levels.

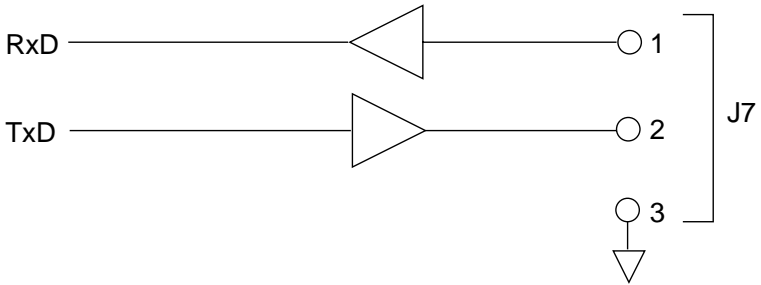


Figure 9–1. J7 RS-232 DCE Configuration.

PROGRAMMABLE REGISTERS

Six registers are used for communication with the SCU. The Serial Transmit Buffer (STB) and Serial Receive Buffer (SRB) store data to be transferred to the serial link and from the serial link, respectively. The Serial Command (SCM) and Serial Mode (SMD) registers define the operating mode. The Serial Interrupt Mask (SIMK) register controls the receive and transmit interrupts. The Serial Status (SST) register provides information on the current state of the transmitter and receiver.

The base I/O address of the SCU registers is defined by the OPHA and SULA registers. OPHA is programmed with the high byte and SULA with the low byte of the 16-bit address; see page 5-6 for details. The address of each register, relative to the base address, is shown in Table 9-1.

Table 9-1
SCU Register Addressing.

Address	Register	Operation	Page Number
Base + 0	Receive Buffer	Read	9-4
Base + 0	Transmit Buffer	Write	9-4
Base + 1	Status Register	Read	9-5
Base + 1	Command Register	Write	9-6
Base + 2	Mode Register	Write	9-7
Base + 3	Interrupt Mask Register	Read/Write	9-8

Serial Status Register (SST)

Figure 9-2 shows the architecture of the SST register, which can be read at any time.

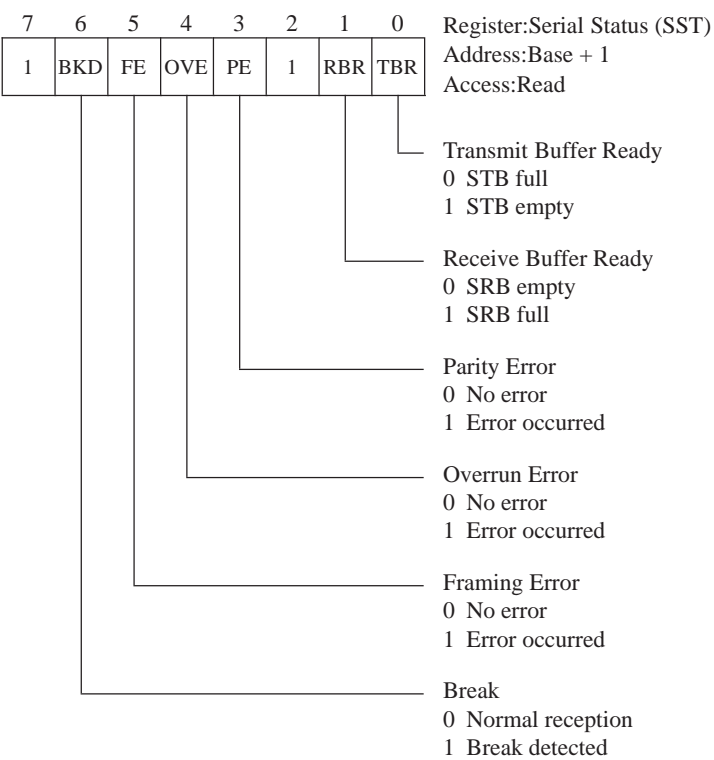


Figure 9-2. Serial Status Register.

Serial Command Register (SCM)

Figure 9-2 illustrates the SCM register bit map. The SCU is configured with the SCM and the SMD registers. The SCM register includes the functions that are most likely to be modified during operation. The SMD register will more than likely be programmed only once for initialization.

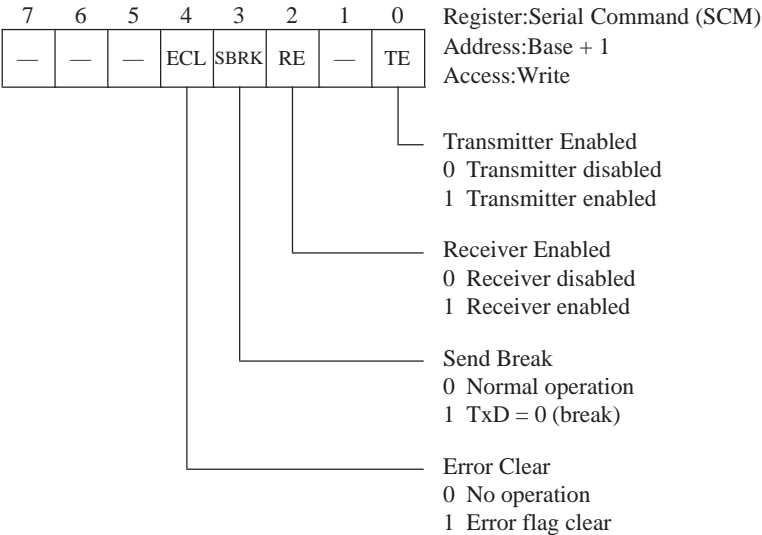


Figure 9-3. Serial Command Register.

Serial Mode Register (SMD)

The format for the SMD register is given in Figure 9-4. The SMD register includes all the functions that are not likely to change after they have been initialized. Bits 1 and 2 combine to define the clock divisor for the baud rate. Programming the baud rate is defined in more detail on page 9-9.

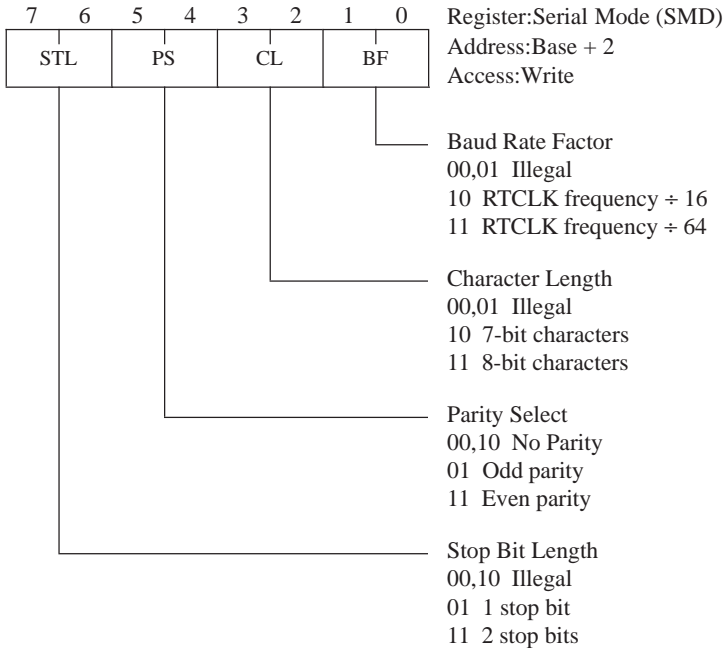


Figure 9-4. Serial Mode Register.

Serial Interrupt Mask Register (SIMK)

The SCU is capable of interrupting the CPU when a character is received into the Serial Receive Buffer or transmitted out of the Serial Transmit Buffer. The SIMK register includes two programmable bits, as illustrated in Figure 9-5, to enable the interrupts. Setting the RM bit to a logical 1 prevents the SCU from generating an interrupt when a character is received. A logical 0 in the RM bit enables the interrupt. The TM bit provides the same control for transmitted characters.

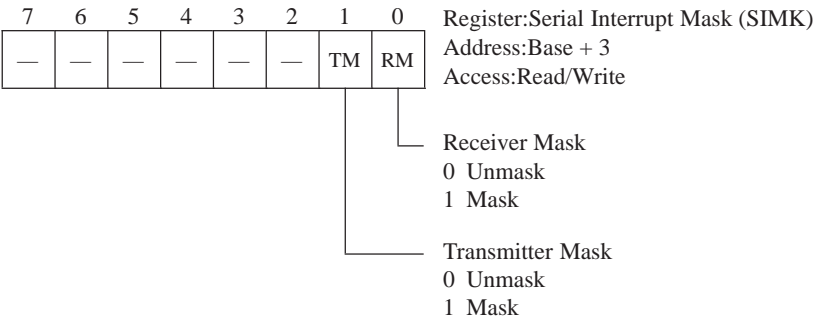


Figure 9–5. Serial Interrupt Mask Register.

BAUD RATE

The SCU baud rate is determined by the output of Counter/Timer 1. Counter/Timer 1 must be initialized for a specific mode of operation and programmed with a count that defines the required baud rate. The following discussion explains the initialization and how to calculate the count. Note that 38.4 Kbaud is not supported except in applications that can accept a baud rate not within 4% of the nominal rate (for example, ZT 8802 to ZT 8802).

To use Counter/Timer 1 as a baud rate generator, the TCKS register must specify that Counter/Timer 1 has an internal clock with a divisor determined by the formula below. The TCU Counter/Timer Mode register must also be programmed to configure Counter/Timer 1 for binary operation as a square wave generator (Mode 3). Different baud rates are obtained by programming Counter/Timer 1 with different counts. The relationship between the baud rate and the count is given below:

$$\text{Count} = \text{V40 Clock} / (\text{Prescale} \times \text{Baud Rate} \times \text{Baud Factor})$$

This formula includes all the factors that determine the baud rate as a function of the V40 clock. The best way to understand the formula is to trace the clock signal from the V40 clock to the SCU. The V40 clock starts with a value of 8.0×10^6 Hz. This frequency is prescaled as defined by the PS bits in the TCKS V40 configuration register. At this point the clock is input to the counter/timers. Counter/Timer 1 further divides the signal by the programmed count before it is input to the SCU. The last division is accomplished internally to the SCU by the baud factor programmed in the BF bits of the Serial Mode register.

As an example, assume the following parameters are specified:

V40 Clock = 8.0×10^6 Hz

Prescale = divide by two (TCKS PS bits = 00)

Baud Rate = 9600

Baud Factor = divide by 16 (SMD BF bits = 10)

The calculation to determine the count value to be programmed into Counter/Timer 1 is shown below to be 26. Please note that the calculated count and the programmed count differ by 0.16 percent. To guarantee proper operation, the percentage difference must never be greater than four. Table 9-2 lists the count values to be programmed into Counter/Timer 1 to generate the more common baud rates.

$$\text{Count} = 8 \times 10^6 / (2 \times 9600 \times 16) = 26.04$$

Table 9-2
ZT 8802 Baud Rate Counts.

BAUD RATE	COUNT DEC/HEX ^[1]	
	Baud Factor Divided By:	
	16	64
110	2273/08E1	568/0238
150	1667/0683	417/01A1
300	833/0341	208/00D0
600	417/01A1	104/0068
1200	208/00D0	52/0034
2400	104/0068	26/001A
4800	52/0034	13/000D
9600	26/001A	[2]
19200	13/000D	[2]
38400	[2]	[2]

[1]The count values listed assume the PS bits of the TCKS V40 configuration register are programmed with logical 0s.

[2]This combination exceeds the recommended 4% error allowance.

If the counter/timers are driven with the external TCLK, instead of the V40 clock, the formula for calculating the value to be loaded into Counter/Timer 1 is given below. Note that the frequency of the external clock signal is not dependent on the PS bit of the TCKS register.

$$\text{Count} = \text{External Clock} / (\text{Baud Rate} \times \text{Baud Factor})$$

ADDITIONAL INFORMATION

Refer to the NEC *16-Bit V Series Microprocessor* data book for more information on the V40 serial controller.

Chapter 10

SERIAL COMMUNICATIONS (16C452)

Contents	Page
OVERVIEW	10-2
SERIAL COMMUNICATIONS PROTOCOL	10-3
SERIAL INTERFACE (RS-232-C)	10-7
Signal Definitions	10-9
Clear-To-Send Inputs (CTS0*, CTS1*)	10-9
Data-Set-Ready (DSR0*, DSR1*)	10-9
Data-Terminal-Ready (DTR0*, DTR1*)	10-10
Serial Channel Interrupt Outputs (INT0, INT1)	10-10
Ring Indicator Inputs (RI0*, RI1*)	10-11
Receive Line Signal Detect (RLSD0*, RLSD1*)	10-11
Reset Control (RESET*)	10-12
Request-To-Send (RTS0*, RTS1*)	10-13
Serial Data Inputs (SIN0, SIN1)	10-13
Serial Data Outputs (SOUT0, SOUT1)	10-13
SERIAL REGISTERS	10-14
Transmit and Receive Buffer Registers	10-18
Scratchpad Register	10-18
Line Control Register	10-19
Baud Rate Generator	10-22
Line Status Register	10-25
Interrupt ID Register	10-27
Interrupt Enable Register	10-29
Modem Control Register	10-30
Modem Status Register	10-32

OVERVIEW

This chapter describes the two 16C450-equivalent serial ports available on the ZT 8802. They are referred to as COM1 and COM2 in DOS systems and ports 1 and 2 in non-DOS systems. These two serial ports are contained within the VL 16C452 Communications Element from VLSI Technologies. COM1 is available at J4 and COM2 is available at J3.

The first section of this chapter details an overview of the software communications priority scheme, with a basic outline of the protocol between two serial devices. The second section details the serial port signals, and the third section contains the register addresses and definitions.

SERIAL COMMUNICATIONS PROTOCOL

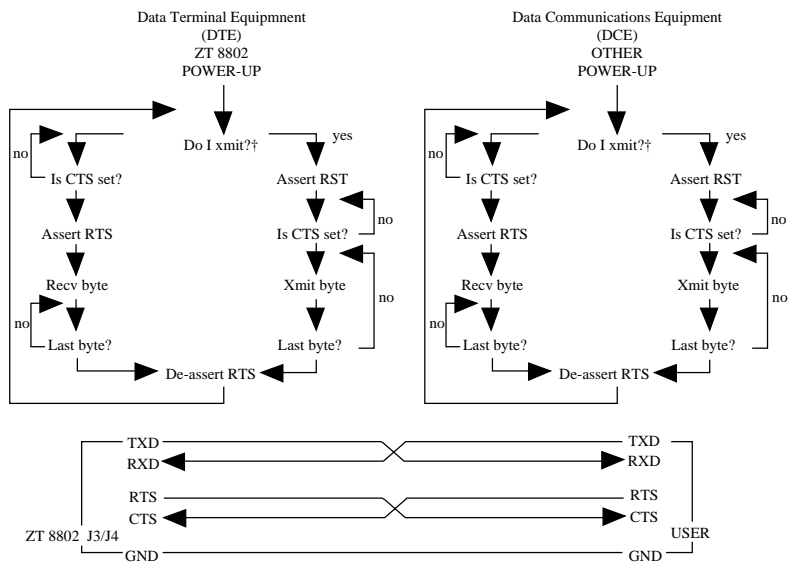
The following paragraphs describe the functioning of a serial data link between the ZT 8802 and a terminal or other computer. The ZT 8802 is shipped configured as Data Terminal Equipment (DTE) for both serial ports 1 and 2. DTE is normally the host end of the serial link which connects to a modem or mouse that is configured as Data Communications Equipment (DCE).

One general principle for DCE and DTE: if one piece of equipment is configured in one sense (DCE, for example), the equipment with which it is communicating must be configured in the opposite sense (DTE, for example). The ZT 8802 employs a cable that converts the frontplane serial signals to a female 9-pin D-type connector as defined in the *EIA Standard RS-232-C, Section 3.1*, formally defining the ZT 8802 as DTE.

The protocol shown in Figure 10-1 on page 10-4 could be used to link the ZT 8802 to a host computer. Only the Request-To-Send (RTS) and Clear-To-Send (CTS) handshake lines are used, although both RTS/CTS and Data-Set-Ready (DSR) and Data-Terminal-Ready (DTR) are provided.

Both the transmit loop and the receive loop test for the CTS signal and assert the RTS signal, but in opposite order from each other. In the receive loop, CTS is set as a result of the other device asserting RTS, indicating that the other device wants to transmit. Notice that RTS and CTS are crossed at the interface. The ZT 8802 can monitor the CTS signal via the Modem Status register, bit 4. The software should loop until CTS is set. Refer to Table 10-3 on page 10-16 for a definition of all the serial port registers.

In the transmit loop, CTS is tested until set, indicating that the other device is ready to receive data. CTS occurs when the transmitter-to-be sends RTS, asking the receiver-to-be to prepare to receive. The transmit loop starts by asserting RTS, telling the other device to prepare to receive. The other device signals it is okay to transmit by asserting CTS. The RTS line on the ZT 8802 is asserted by writing to the Modem Control register, setting bit 1.



†A communications priority scheme must be provided at this junction to prevent both devices from transmitting at the same instant. The system designer must assign either the DCE or the DTE to transmit only after having received a message indicating it is okay to transmit.

Figure 10-1. Establishing Serial Communications.

When transmitting data, the system software must be sure the transmitter output buffer is empty (the previous data, if any, having been sent to the receiver). This is accomplished on the ZT 8802 by reading the Transmitter Holding Register Empty status found in the Line Status register, bit 5. When receiving data, the software must be sure the receiver buffer is full but not overrun. In the ZT 8802 this is accomplished by reading the Data Ready status found in the Line Status register, bit 0.

Both the transmit and the receive loop test for the last byte by specifying a data string length and/or an End-Of-String (EOS) character in the software. When the data transfer is complete, the RTS line is deasserted. When the ZT 8802 was receiving data, this action told the DCE it was no longer "clear to send" data. When the ZT 8802 was transmitting, it told the DCE that the ZT 8802 was no longer "requesting to send" data. The software should loop back and repeat the process again.

The ZT 8802 COM1 and COM2 are shipped configured as DTE. If you prefer to use only a "three-wire" serial interface (that is, TxD, RxD, and ground), the ZT 8802 can be used without the RTS and CTS lines. Both serial devices appear to be always ready to transmit and receive.

System software is more complicated this way because both serial devices have to keep a sharp lookout for transmitter and receiver buffer activity. This is best accomplished on the ZT 8802 by using interrupts. Details on the different aspects of interrupt usage are discussed later in the serial interrupt register sections of this chapter.

Functionally, whenever an incoming data stream fills the receiver buffer, an interrupt should be generated, telling the CPU to hurry up and read the receiver buffer for incoming data. The transmit process is less critical, but we recommend that the processor be interrupted when the transmitter buffer is ready to be loaded again by the CPU; that is, when the previous transmission is complete.

If the software for handling the serial control lines has already been designed and you want to implement the three-wire serial link, the ZT 8802 can be jumpered to loop RTS back to CTS, and DTR to DSR. This is done by connecting these signals to each other in the cable.

SERIAL INTERFACE (RS-232-C)

The ZT 8802 provides two high-speed RS-232-C serial ports. These use the same programming architecture and pin definitions as the popular 8250 from Western Digital. The serial ports are contained within the 16C452 Communications Element.

Figure 10-2 on page 10-8 shows the block diagram of a 16C452 serial port. This device performs serial-to-parallel conversion on data characters received from a peripheral device or a modem and parallel-to-serial conversion on data characters received from the STD CPU.

The STD CPU can read the complete status of the serial interface at any time. Status information reported includes the type and condition of the transfer operation being performed, as well as any error conditions.

Each serial port features an on-board baud rate generator. The baud rate is controlled by software and is usually determined by the initialization routine. The baud rate generator is capable of dividing the timing reference clock (1.84 MHz) input by 1 to 65,536, to produce a 16x clock for driving the internal transmitter logic. This 16x clock also drives the receiver logic.

The serial interface also includes Clear-To-Send (CTS), Request-To-Send (RTS), Data-Set-Ready (DSR), Data-Terminal-Ready (DTR), and modem-control capability.

The ZT 8802 provides fully buffered RS-232-C serial data and control lines via two connectors, supplying the ± 12 V swing needed to meet RS-232-C driver requirements. The RS-232-C buffers meet the RS-232-C standard for signal conditioning for the recommended 50-foot cables. Longer cables are permissible provided the resulting load capacitance does not exceed 2500 pF. The ZT 8802 uses a Maxim MAX249 serial line driver/receiver for creating the RS-232-C compatible signals.

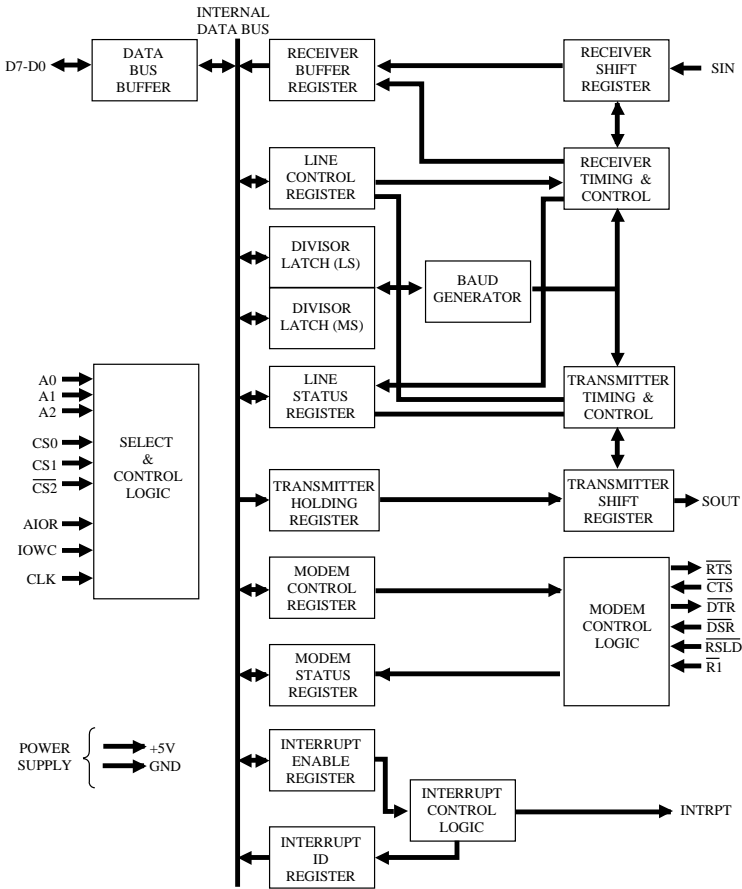


Figure 10–2. 16C452 Serial Port Block Diagram.

Signal Definitions

The following is a description of each of the 16C452 signal inputs and outputs in the signal name. The 0 (zero) refers to serial port 1 (COM1) and the 1 (one) to serial port 2 (COM2).

Clear-To-Send Inputs (CTS0*, CTS1*)

The logical state of each CTS* pin is reflected in the CTS bit (bit 4) of the Modem Status register (MSR). A change of state in the CTS* pin since the previous reading of the associated MSR in each serial port causes the setting of the DCTS bit (bit 0) in the respective MSR. When CTS* is active (low), the modem is indicating that data on the associated serial output (SOUT) can be transmitted.

Data-Set-Ready (DSR0*, DSR1*)

The logical state of the DSR* pins is reflected in the DSR bit (bit 5) in the MSR of the associated serial port. The DDSR bit, which is bit 1 in the MSR, indicates whether the associated DSR* pin has changed state since the previous reading of the MSR. When a DSR* pin is active (low), its modem is indicating that it is ready to exchange data with the associated UART.

Data-Terminal-Ready (DTR0*, DTR1*)

Each DTR* pin can be set active (low) by writing a logical 1 to the DTR bit (bit 0) in the Modem Control register (MCR) of its associated UART. This signal is cleared (high) by writing a logical 0 to the DTR bit in the MCR or whenever a reset occurs. When active, the DTR* pin indicates that its UART is ready to receive data.

Serial Channel Interrupt Outputs (INT0, INT1)

Each serial channel interrupt goes active (high) when one of the following interrupt sources has an active condition and is enabled by the Interrupt Enable register (IER) of its associated channel: Receiver Error flag, Received Data Available, Transmitter Holding Register Empty (THRE), and Modem Status. The interrupt is reset low upon appropriate service or a reset operation.

Factory default assigns INT0 to Interrupt Request level 4 (IR4) and INT1 to Interrupt Request level 3 (IR3) at the 8259A Programmable Interrupt Controller on board. These correspond to COM1 and COM2 in a DOS system, respectively, identical to the interrupt levels in an IBM PC or equivalent.

Ring Indicator Inputs (RI0*, RI1*)

When active (low), RI* indicates that a telephone ringing signal has been received by the modem or data set. The RI* signal is a modem control input whose condition is tested by reading the RI*, bit 6, of the associated UART's MSR. The MSR output bit TERI, bit 2, indicates whether the RI* input has changed from high to low since the previous reading of the same MSR. If the interrupt is enabled (indicated by the Interrupt Enable register bit 3 being set to 1), and RI* changes from high to low, an interrupt is generated for that UART.

Receive Line Signal Detect (RLSD0*, RLSD1*)

When active (low), RLSD* (sometimes referred to as Data Carrier Detect, or DCD) indicates that the data carrier has been detected by the modem or data set. RLSD* is a modem input whose condition can be tested by reading the RLSD (DCD) bit (bit 7) of the MSR. The DRLSD (DDCD) bit (bit 3) of the MSR indicates whether the RLSD input has changed since the previous reading of the MSR. RLSD* has no effect on the receiver. If the RLSD* changes state with the modem status interrupt enabled, an interrupt is generated.

Reset Control (RESET*)

The ZT 8802 contains power-up and pushbutton reset circuitry that drives the RESET* input signal at the serial ports. The reset forces the serial ports into an idle mode in which all serial data activities are suspended. The Modem Control register (MCR) and its associated outputs are cleared. The Line Status register (LSR) is cleared except for the THRE and TEMT bits, which are set. All functions of the device remain in an idle state until programmed to resume serial data activities. Table 10-1 illustrates the effect of reset on the serial ports.

Table 10-1
16C452 Reset State.

Register/Signal	Reset Control	Reset
Interrupt Enable Register	Reset	All bits low (0-3 forced and 4-7 permanent)
Interrupt Identification Register	Reset	Bit 0 is high, bits 1 & 2 low bits 3-7 permanently low
Line Control Register	Reset	All bits low
Modem Control Register	Reset	All bits low
Line Status Register	Reset	All bits low, except bits 5 and 6 are high
Modem Status Register	Reset	Bits 0-3 low, bits 4-7 input signal
SOUT	Reset	High
Intrpt (RCVR Errs)	Read LSR, Reset	Low
Intrpt (RCVR Data Ready)	Read RBR, Reset	Low
Intrpt (THRE)	Read IIR, Write THR, Reset	Low
Intrpt (Modem Status Changes)	Read MSR, Reset	Low
-Out2	Reset	High
-RTS	Reset	High
-DTR	Reset	High
-Out1	Reset	High

Request-To-Send (RTS0*, RTS1*)

The RTS* pin is set active (low) by writing a logical 1 to bit 1 of the associated UART's Modem Control register. Both RTS* pins are disabled (set high) by reset. The RTS* signal on each UART is used to enable the modem. When active, an RTS* pin indicates that its UART has data ready to transmit. In half-duplex operations, RTS* is used to control the direction of the line.

Serial Data Inputs (SIN0, SIN1)

The serial data inputs move information from the communication line or modem to the UART receiver circuits. A mark (1) is high; a space (0) is low. Data on serial data inputs is disabled when operating in the loopback mode.

Serial Data Outputs (SOUT0, SOUT1)

These lines are the serial data outputs from the UARTs' transmitter circuitry. A mark (1) is a logical 1 (high); a space (0) is a logical 0 (low). Each SOUT is held in the mark condition when the transmitter is disabled, when RESET* is active (low), when the Transmitter register is empty, or when in the loopback mode.

SERIAL REGISTERS

This section describes the individual UART registers.

You may access or control any of the serial registers summarized in Table 10-3 on page 10-16. The registers are used to control the serial ports' operation and to transmit or receive data. There is a complete set of these registers for each UART.

The base I/O address of each UART is 03F8h for serial port 1 (COM1) and 02F8h for serial port 2 (COM2). The offset of each register from the base is shown in Table 10-3 on pages 10-16 and 10-17 and in the "I/O Port Assignments" table on page 10-15.

Note that the state of the Divisor Latch Access Bit (DLAB), which is the most significant bit of the Line Control register, affects the selection of certain UART registers. (See page 10-21 for a full description.) DLAB is reset low when the ZT 8802 is reset or powered up; the DLAB must be set high by the system software to access the baud rate generator divisor latches, and reset low again to access the remaining UART registers.

Table 10-2
ZT 8802 I/O Port Assignments.

I/O Port Base +	I/O Address	I/O Read Register	I/O Write Register
0	3F8h	Data Buffer Ch1	Data Buffer Ch1
1	3F9h	Intr. Enable Ch1	Intr. Enable Ch1
2	3FAh	Intr. Ident. Ch1	---
3	3FBh	Line Control Ch1	Line Control Ch1
4	3FCh	Modem Cntrl. Ch1	Modem Cntrl. Ch1
5	3FDh	Line Status Ch1	Line Status Ch1
6	3FEh	Modem Status Ch1	---
7	3FFh	---	---
0	2F8h	Data Buffer Ch2	Data Buffer Ch2
1	2F9h	Intr. Enable Ch2	Intr. Enable Ch2
2	2FAh	Intr. Ident. Ch2	---
3	2FBh	Line Control Ch2	Line Control Ch2
4	2FCh	Modem Cntrl. Ch2	Modem Cntrl. Ch2
5	2FDh	Line Status Ch2	Line Status Ch2
6	2FEh	Modem Status Ch2	---
7	2FFh	---	---

Note: Serial port 1 base is 3F8h, serial port 2 base is 2F8h.

Table 10-3

16C452 Addressable Registers Summary.

Register Address					
Bit No.	0 DLAB = 0	0 DLAB = 0	1 DLAB = 0	2	3
	Receive Buffer (Read Only)	Transmit Buffer (Write Only)	Interrupt Enable Register	Interrupt Identify Register (Read Only)	Line Control Register
	RBR	THR	IER	IIR	LCR
0	Data Bit 0*	Data Bit 0	Enable Receive Buffer Full Interrupt (MSI)	"0" if Interrupt Pending	Word Length Select Bit 0 (WLS0)
1	Data Bit 1	Data Bit 1	Enable Transmit Buffer Empty Interrupt (RSI)	Interrupt ID Bit (0)	Word Length Select Bit 1 (WLS1)
2	Data Bit 2	Data Bit 2	Enable Receive Status Interrupt (TBI)	Interrupt ID Bit (1)	Number of Stop Bits (STB)
3	Data Bit 3	Data Bit 3	Enable Modem Status Interrupt (RBI)	0	Parity Enable (PEN)
4	Data Bit 4	Data Bit 4	0	0	Even Parity Select (EPS)
5	Data Bit 5	Data Bit 5	0	0	Stick Parity
6	Data Bit 6	Data Bit 6	0	0	Set Break
7	Data Bit 7	Data Bit 7	0	0	Divisor Latch Access Bit (DLAB)

*Bit 0 is the least significant bit. It is the first bit serially transmitted or received.

Table 10-3
16C452 Addressable Registers Summary (continued).

Register Address						
Bit No.	4	5	6	7	0 DLAB = 1	1 DLAB = 1
	Modem Control Register	Line Status Register	Modem Status Register	Scratchpad Register	Divisor Latch (LSB)	Divisor Latch (MSB)
	MCR	LSR	MSR	SCR	DLL	DLM
0	Data Terminal Ready (DTR)	Data Ready (DR)	Delta Clear to Send (DCTS)	SCR0	D0	D8
1	Request to Send (RTS)	Overrun Error (OE)	Delta Data Set Ready (DDSR)	SCR1	D1	D9
2	OUT1 (Not available externally)	Parity Error (PE)	Trailing Edge Ring Indicator (TERI)	SCR2	D2	D10
3	OUT2 (Interrupt output enable)	Framing Error (FE)	Delta Data Carrier Detect (DDCD)	SCR3	D3	D11
4	Loop	Break Interrupt (BI)	Clear to Send (CTS)	SCR4	D4	D12
5	0	Transmit Holding Register (THRE)	Data Set Ready (DSR)	SCR5	D5	D13
6	0	Transmit Empty (TEMT)	Ring Indicator (RI)	SCR6	D6	D14
7	0	0	Data Carrier Detect (DCD)	SCR7	D7	D15

Transmit and Receive Buffer Registers

The Transmitter Buffer register and Receiver Buffer register are data registers holding from five to eight bits of data. If less than eight data bits are transmitted, data is right justified to the LSB. Bit 0 of a data word is always the first serial data bit received and transmitted.

The 16C452 data registers are double-buffered so that read and write operations can be performed at the same time the UART is performing the parallel-to-serial and serial-to-parallel conversion.

Scratchpad Register

The Scratchpad register (SCR) is an 8-bit Read/Write register that has no effect on either serial channel. It is intended to be used by the programmer to hold data temporarily.

Line Control Register

(2FBh, 3FBh; R/W)

Use the Line Control register (LCR) to specify the format of the asynchronous data communications exchange. In addition to controlling the format, you may retrieve the contents of the Line Control register for inspection. This feature simplifies system programming and eliminates the need for storing line characteristics in system memory. Contents of the LCR are included in Table 10-3 on page 10-16 and are described below.

Bits 0 and 1 These two bits specify the number of bits in each transmitted or received serial character. The encoding of bits 0 and 1 is as follows:

Bit 1	Bit 0	Word Length
0	0	5 bits
0	1	6 bits
1	0	7 bits
1	1	8 bits

- Bit 2* Bit 2 specifies the number of stop bits in the transmitted or received serial character. If bit 2 is a logical 0, one stop bit is generated or checked in the transmit or receive data, respectively. If bit 2 is a logical 1 when a 5-bit word length is selected via bits 0 and 1, 1½ stop bits are generated or checked. If bit 2 is a logical 1 when either a 6-, 7-, or 8-bit word length is selected, two stop bits are generated or checked. The receiver checks for two stop bits if programmed.
- Bit 3* This bit is the Parity Enable bit. When bit 3 is a logical 1, a parity bit is generated (transmit data) or checked (receive data) between the last data word bit and stop bit of the serial data.
- Bit 4* This is the Even Parity Select bit. When bit 3 is a logical 1 and bit 4 is a logical 0, an odd number of logical 1s is transmitted or checked in the data word bits and parity bit. When bit 3 is a logical 1 and bit 4 is a logical 1, an even number of bits is transmitted or checked.
- Bit 5* Bit 5 is the Stick Parity bit. When bit 3 is a logical 1 and bit 5 is a logical 1, the parity bit is transmitted and then detected by the receiver in the opposite state indicated by bit 4. This allows you to force parity to a known state and allows the receiver to check the parity bit in a known state.

Bit 6 This is the Set Break Control bit. When bit 6 is a logical 1, the serial output (SOUT) is forced to the spacing (logical 0) state until reset by a low-level bit 6, regardless of other transmitter activity. This allows the CPU to alert a terminal in a computer communications system and has no effect on the transmitter logic. If the following sequence is used, no erroneous or extraneous characters are transmitted because of the break.

1. Load an all "0s" pad character in response to THRE.
2. Set Break in response to the next THRE.
3. Wait for the transmitter to be idle (TEMT = 1), and clear break when normal transmission has been restored.

Bit 7 Bit 7 is the Divisor Latch Access Bit (DLAB). It must be set high (logical 1) to access the divisor latches of the baud rate generator during any read or write operation. DLAB must be set low (logical 0) to access the Receiver Buffer, the Transmitter Holding register, or the Interrupt Enable register.

Baud Rate Generator

The serial baud rate generator takes the clock input (1.8432 MHz or 8 MHz) and divides it by any divisor from 1 to 65,536. The output frequency of the baud rate generator is 16 times the baud rate.

Two 8-bit latches store the divisor in a 16-bit binary format. These divisor latches must be loaded during initialization in order to ensure desired operation of the baud rate generator.

When loading either of the divisor latches, a 16-bit baud counter immediately becomes effective. Table 10-4 illustrates the use of the baud generator with the on-board 1.8432 MHz oscillator. Table 10-5 illustrates the use of the baud generator with the on-board 8 MHz oscillator.

The baud rate divisor was calculated from the following relationship:

$$D = F / (16 \times B)$$

where: D is the divisor value

F is the clock frequency (1.8432 MHz), and

B is the baud rate.

Note: The maximum operating frequency of the baud generator is 1.8432 MHz. However, when using divisors of 5 and below, the maximum frequency is equal to the divisor in MHz (a "1" divisor = 1 MHz maximum frequency). In no case should the data rate be greater than 56 Kbaud.

Table 10-4
Baud Rates, 1.8432 MHz Clock.

Baud Rates Using 1.8432 MHz Clock (F)		
Baud Rate (B)	Divisor (D)	% Error
50	2304	--
75	1536	--
110	1047	0.026
134.5	857	0.058
150	768	--
300	384	--
600	192	--
1200	96	--
1800	64	--
2000	58	.69
2400	48	--
3600	32	--
4800	24	--
7200	16	--
9600	12	--
19200	6	--
38400	3	--
56000	2	2.86

Table 10-5
Baud Rates, 8 MHz Clock.

Baud Rates Using 8.0 MHz Clock (F)		
Baud Rate (B)	Divisor (D)	% Error
50	1000	--
75	6667	0.005
110	4545	0.010
134.5	3717	0.013
150	3333	0.010
300	1667	0.020
600	833	0.040
1200	417	0.080
1800	277	0.080
2000	250	--
2400	208	0.160
3600	139	0.080
4800	104	0.160
7200	69	0.644
9600	52	0.160
19200	26	0.160
38400	13	0.160
56000	9	0.790
128000†	4	2.344
256000†	2	2.344
512000†	1	2.400

† Not recommended

Line Status Register

(2FDh, 3FDh; R/W)

This 8-bit register provides status information to the CPU concerning the data transfer. Reading the Line Status register (LSR) clears bits 1 through 4 (OE, PE, FE, and BI). The contents of the LSR are included in Table 10-3 on pages 10-16 and 10-17, and a description follows.

- Bit 0* This bit is the receiver Data Ready (DR) indicator. Bit 0 is set to logical 1 whenever a complete incoming character has been received and transferred into the Receiver Buffer register. Bit 0 may be reset to logical 0 either by the CPU reading the data in the Receiver Buffer register or by writing a logical 0 into the LSR from the CPU.
- Bit 1* This bit is the Overrun Error (OE) indicator. It indicates that data in the Receiver Buffer register was not read by the CPU before the next character was transferred into the Receiver Buffer register, thereby destroying the previous character. The OE indicator is reset whenever the CPU reads the contents of the LSR.
- Bit 2* This bit is the Parity Error (PE) indicator. Bit 2 indicates that the received data character does not have the correct even or odd parity, as selected by the Even Parity Select bit. The PE bit is set to logical 1 upon detection of a parity error and is reset to logical 0 whenever the CPU reads the contents of the LSR.
- Bit 3* This bit is the Framing Error (FE) indicator. Bit 3 indicates that the received character did not have a valid stop bit. Bit 3 is set to logical 1 whenever the stop bit following the last data bit or parity bit is detected as a zero bit (spacing level).

Bit 4 This bit is the Break Interrupt (BI) indicator. Bit 4 is set to logical 1 whenever the received data input is held in the spacing (logical 0) state for longer than a full word transmission time (that is, the total time of start bit + data bits + parity + stop bits).

Note: Bits 1 through 4 of the LSR are the error conditions that produce a Receiver Line Status interrupt whenever any of the corresponding conditions are detected.

Bit 5 This bit is the Transmitter Holding Register Empty (THRE) indicator. Bit 5 indicates that the 16C452 is ready to accept a new character for transmission. In addition, this bit causes the 16C452 to issue an interrupt to the CPU when the THRE Interrupt Enable is set high. The THRE bit is set to logical 1 when a character is transferred from the Transmitter Holding register into the Transmitter Shift register. The bit is reset to logical 0 concurrently with the loading of the Transmitter Holding register by the CPU.

Bit 6 This bit is the Transmitter Shift Register Empty (TEMT) indicator. Bit 6 is set to logical 1 whenever the Transmitter Shift register is idle. It is reset to logical 0 upon a data transfer from the Transmitter Holding register to the Transmitter Shift register and remains low until the character is transferred out of SOUT. Bit 6 is a read-only bit.

Bit 7 This bit is permanently set to logical 0.

Interrupt ID Register

(2FAh, 3FAh; R)

The Interrupt Identification register (IIR) stores an identification code or "ID" of pending interrupts. In order to provide minimum software overhead during data character transfers, the serial hardware prioritizes interrupts into four levels: Receiver Line Status (priority 1), Received Data Ready (priority 2), Transmitter Holding Register Empty (priority 3), and Modem Status (priority 4). Refer to Table 10-6 below.

Table 10-6
16C452 Interrupt Control Functions.

Interrupt Identification Register			Interrupt Set and Reset Functions			
Bit 2	Bit 1	Bit 0	Priority Level	Interrupt Flag	Interrupt Source	Interrupt Reset Control
0	0	1	—	None	None	—
1	1	0	Highest	Receiver Line Status	Overrun Error or Parity Error or Framing Error or Break Interrupt	Reading the Line Status Register
1	0	0	Second	Received Data Available	Receiver Data Available	Reading The Receiver Buffer Register
0	1	0	Third	Transmitter Holding Register Empty	Transmitter Holding Register Empty	Reading the IIR Register (if source of interrupt) or Writing into the Transmitter Holding Register
0	0	0	Fourth	Modem Status	Clear to Send or Data Set Ready or Ring Indicator or Received Line Signal Detect	Reading the Modem Status Register

Information stored in the IIR indicates that a prioritized interrupt is pending. The source of the interrupt is also indicated. The IIR, when addressed during chip-select time, freezes the highest priority interrupt pending, and no other interrupts are acknowledged until the particular interrupt is serviced by the CPU. The contents of the IIR are indicated in Table 10-3 on pages 10-16 and 10-17 and are described below.

Bit 0 This bit can be used in either a hardwired prioritized or a polled environment to indicate whether an interrupt is pending. When bit 0 is a logical 0, an interrupt is pending and the IIR contents may be used as a pointer to the appropriate interrupt service routine. When bit 0 is a logical 1, no interrupt is pending and polling (if used) continues.

Bits 1 and 2 These two bits of the IIR are used to identify the highest priority interrupt pending, as indicated in Table 10-6, "16C452 Interrupt Control Functions."

Bits 3 - 7 These five bits of the IIR are always logical 0.

Interrupt Enable Register

(2F9h, 3F9h; R/W)

The Interrupt Enable register (IER) enables the four interrupt sources of the serial interface to separately activate the on-board interrupt hardware. It is possible to totally disable the interrupt system by resetting bits 0-3 of the IER. Similarly, by setting the appropriate bits of this register to logical 1, selected interrupts can be enabled. Keep in mind that the Modem Control register (MCR) bit 3, the interrupt output enable bit, must be set for interrupts to occur.

Disabling the interrupt system inhibits the Interrupt Identification register (IIR) and the active (high) INTRPT output from the chip. All other system functions operate in their normal manner, including the setting of the Line Status and Modem Status registers. Contents of the IER are included in Table 10-3 on pages 10-16 and 10-17 and are described below.

Bit 0 This bit enables the Received Data Available Interrupt when set to logical 1.

Bit 1 This bit enables the Transmitter Holding Register Empty Interrupt when set to logical 1.

Bit 2 This bit enables the Receiver Line Status Interrupt when set to logical 1.

Bit 3 This bit enables the Modem Status Interrupt when set to logical 1.

Bits 4 - 7 These four bits are always logical 0.

Modem Control Register

(2FCh, 3FCh; R/W)

The Modem Control register (MCR) controls the interface with the modem or data set (or a peripheral device emulating a modem). The contents of the MCR are included in Table 10-3 on pages 10-16 and 10-17 and are described below. The RTS* and DTR* outputs are directly controlled by their control bits in this register. A high written to these bits asserts the signal active (low) at the output.

- Bit 0* This bit controls the Data-Terminal-Ready (DTR*) output. Setting this bit to 1 asserts the DTR* output active (low). By resetting this bit to 0, the DTR* output is set inactive (high).
- Bit 1* This bit controls the Request-To-Send (RTS*) output. When bit 1 is set to logical 1, the RTS* output is set active (low). When bit 1 is reset to logical 0, the RTS* output is set inactive (high).
- Bit 2* This bit would normally control the Output 1 (OUT1) signal, which is an auxiliary user-designated output. The VL 16C452 implementation of the 16C452 serial port OUT1 is not connected to an output pin.
- Bit 3* This bit normally controls the Output 2 (OUT2) signal on a 16C450, which is an auxiliary user-designated output. In this implementation of the 16C452, this signal controls the output enable to the buffered interrupt request from the associated interrupt controller. Writing a 1 to this bit enables the interrupt output, and writing a 0 three-states it. This is implemented identically to the IBM PC/XT serial port, making these serial ports fully IBM compatible. We recommend that this bit be set to 1 prior to enabling the associated interrupt input at the 8259 Interrupt Controller on board.

Bit 4 This bit provides a loopback feature for diagnostic testing of the 16C452. When bit 4 is set to logical 1, the following occurs: the transmitter Serial Output (SOUT) is set to the marking (logical 1) state, the receiver Serial Input (SIN) is disconnected, the output of the Transmitter Shift register is "looped back" into the Receiver Shift register input, the four Modem Control inputs (CTS, DSR*, RLSD*, and RI*) are disconnected, and the four Modem Control outputs (DTR*, RTS*, OUT1, and OUT2) are internally connected to the four Modem Control inputs. The Modem Control outputs DTR* and RTS* are set to their inactive state (high).

In the diagnostic mode, data that is transmitted is immediately received. This feature allows the processor to verify the transmit-data and receive-data paths of the 16C452. In the diagnostic mode, the receiver and transmitter interrupts are fully operational. The Modem Control Interrupts are also operational, but the interrupts' sources are now the lower four bits of the MCR instead of the four Modem Control inputs. The interrupts are still controlled by the Interrupt Enable register.

The 16C452 interrupt system can be tested by writing into the lower six bits of the Line Status register and the lower four bits of the Modem Status register. Setting any of these bits to logical 1 generates the appropriate interrupt (if enabled). The resetting of these interrupts is the same as in normal 16C452 operation. To return to normal operation, the register must be reprogrammed for normal 16C452 operation and then bit 4 must be reset to logical 0.

Bits 5 - 7 These bits are permanently set to logical 0.

Modem Status Register

(2FEh, 3FEh; R)

The Modem Status register (MSR) provides the current state of the control lines from the modem (or peripheral device) to the CPU. In addition to this current-state information, four bits of the MSR provide change information.

These bits are set to logical 1 whenever a control input from the modem changes state. They are reset to logical 0 whenever the CPU reads the MSR. The contents of the MSR are included in Table 10-3 on pages 10-16 and 10-17 and are described below.

- Bit 0* This bit is the Delta Clear-To-Send (DCTS) indicator. Bit 0 indicates that the CTS input to the chip has changed state since the last time it was read by the CPU.
- Bit 1* This bit is the Delta Data-Set-Ready (DDSR) indicator. Bit 1 indicates that the DSR input to the chip has changed state since the last time it was read by the CPU.
- Bit 2* This bit is the Trailing Edge-of-Ring Indicator (TERI) detector. Bit 2 indicates the RI* input to the chip has changed from an On (logical 1) to an Off (logical 0) since the last time it was read by the CPU.
- Bit 3* This bit is the Delta Received Line Signal Detector (DRLSD) indicator. Bit 3 indicates that the RLSD input to the chip has changed state since the last time it was read by the CPU.
- Bit 4* This bit is the complement of the Clear-To-Send (CTS*) input. When set, it indicates that the modem is ready to receive data from the Serial Channels Transmitter Output (SOUT). If bit 4 of the MCR is set (loopback mode), this bit is equivalent to RTS* in the MCR.

- Bit 5* This bit is the complement of the Data-Set-Ready (DSR*) input. When set, this bit indicates that the modem is ready to provide received data to the serial channel receiver circuitry. When DSR* is active (low), this bit is set to logical 1. If the channel is in the loopback mode, this bit is equivalent to DTR in the MCR.
- Bit 6* This bit is the complement of the Ring Indicator (RI*) input. If the channel is in the loopback mode, this bit doesn't reflect any MCR bit status.
- Bit 7* This bit is the complement of the Received Line Signal Detect (RLSD*) input. If the channel is in the loopback mode, this bit is equivalent to OUT2 of the MCR.

The modem status inputs (-RI, -RLSD, -DSR, and -CTS) reflect or are activated by any change of status. Reading the MSR clears these indications but has no effect on the status bits. The status bits reflect the state of the input pins, regardless of the mask control signals. If a DCTS, DDSR, TERI, or DRLSD is true, and a state change occurs during a read operation (-DISTR), the state change is not indicated in the MSR. If DCTS, DDSR, TERI, or DRLSD is false, and a state change occurs during a read operation, the state change is indicated after the read operation.

For LSR and MSR, the setting of status bits is inhibited during status register read -DISTR operations. If a status condition is generated during a read -DISTR operation, the status bit is not set until the trailing edge of the read -DISTR.

If a status bit is set during a read -DISTR operation, and the same status condition occurs, that status bit is cleared at the trailing edge of the read -DISTR instead of being set again.

WATCHDOG TIMER

Contents	Page
OVERVIEW	11-1
ZT 8802 SPECIFICS	11-2
FUNCTIONAL DESCRIPTION	11-2
Stage Timer	11-3
Stage Delay	11-3
OPERATION	11-4
Reset	11-4
Programming	11-5
Control Port	11-5
External Strobe	11-6

OVERVIEW

The primary function of the watchdog timer is to monitor ZT 8802 operation and take corrective action if the ZT 8802 fails to function as programmed. The watchdog timer is a single stage implementation. When enabled, the watchdog timer must be strobed within one of three configurable time periods or a reset is generated to the V40 and system.

The major features of the watchdog timer are as follows.

- Enabled and disabled through jumper selection
- Armed and strobed through software control

ZT 8802 SPECIFICS

The watchdog timer must be jumper selected before it is operational. If the watchdog timer is selected, bit 2 of the Control Port (FA05h) becomes dedicated to the watchdog timer and cannot be used for general purpose I/O. It is possible to use an external source to strobe the watchdog by using connector J6, pin 48, as an input and not using the Control Port bit 2 to strobe the watchdog. See the programming section on page 11-5 for details.

FUNCTIONAL DESCRIPTION

A functional diagram of the watchdog timer is illustrated below. The diagram includes a timer and a delay for the single stage. The functional blocks are described on the following pages.

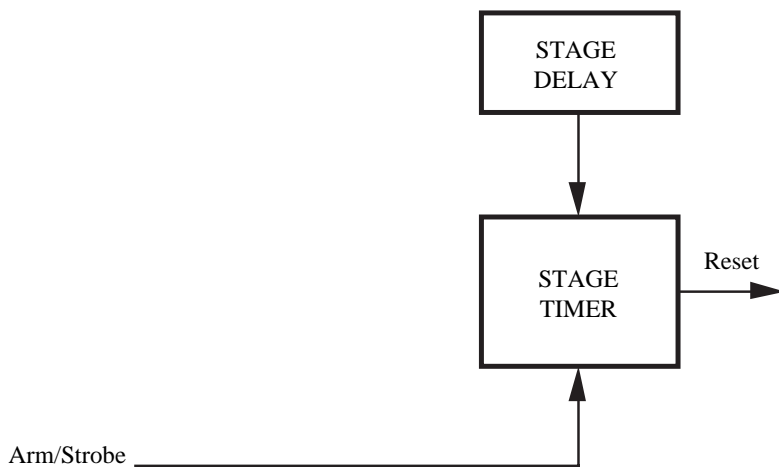


Figure 11-1. Watchdog Timer Functional Diagram.

Stage Timer

The watchdog timer generates a reset to the local CPU if the watchdog timer is jumper selected (W46 IN) and armed, and a strobe does not occur within the time period defined by the Stage Delay functional block.

Stage Delay

The stage delay has a default range of 400 milliseconds minimum and 2000 milliseconds maximum. The minimum delay time means the watchdog timer must be strobed sooner than 400 ms after the last strobe to prevent a local reset. The maximum delay time means it could take up to 2000 ms before the local reset occurs if the watchdog is not strobed. The stage delay is selectable by means of cuttable traces located on the solder side of the board. The available delay ranges are shown in Table 11-1. The lower part of the range is the maximum time software has to strobe the watchdog to guarantee no system reset. The maximum value represents the maximum time the watchdog could take to reset the system.

Table 11-1
Watchdog Timer Stage Delay Options.

CT2	CT1	Min.	Max.	Typ.
IN†	OUT†	400 ms	2000 ms	1200 ms
OUT	OUT	250 ms	1000 ms	600 ms
OUT	IN	50 ms	250 ms	150 ms
IN	IN	NOT ALLOWED		

† Default. Other options are factory special orders.

OPERATION

In operation, the local CPU is programmed to strobe the watchdog timer at a periodic rate less than the stage time delay. If the local CPU fails to operate as programmed, a local reset is generated.

Reset

The watchdog timer is disarmed during and after both a power up and reset condition.

The watchdog timer generates a local reset if allowed to time out. This reset period lasts for up to one second.

Programming

Control Port

The watchdog timer is armed and strobed with bit 2 of the control port at FA05h. Jumper W46 must be in for watchdog operation. The watchdog timer is armed with the following programming sequence:

1. Read the control port at FA05h with interrupts disabled.
2. Logically OR this value with a 4h (set bit 2).
3. Rewrite this new value to port FA05h to arm the watchdog timer. A reset occurs unless the watchdog is strobed before the stage delay time (min.) expires.

The watchdog timer is strobed with the following programming sequence.

1. Read the control port at FA05h with interrupts disabled.
2. Logically AND this value with an FBh (clear bit 2).
3. Rewrite this new value to port FA05h to strobe the watchdog timer.
4. Logically OR the previous value written in the step above with a 4h (set bit 2).
5. Rewrite this new value to port FA05h to arm the watchdog timer again.

External Strobe

The watchdog timer may be set up to be strobed on an external event. If this is desired, Control Port bit 2 must remain cleared, which is the default after reset.

To use an external source, connector J6, pin 48, is used to connect to the watchdog. W46 must be installed. The external source must drive this pin to 0 V to arm the watchdog. It then must drive this input to 5 V and then 0 V before the minimum time selected by CT2 and CT1 to strobe the watchdog and avoid reset.

Chapter 12

SBX EXPANSION MODULE

Contents	Page
OVERVIEW	12-2
Features	12-2
ZT 8802 SPECIFICS	12-3
INSTALLATION	12-4

OVERVIEW

The SBX expansion module provides a means to expand the I/O capabilities of the ZT 8802. The expansion module interface is electrically, mechanically, and functionally compatible with the Intel *iSBX MULTIMODULE™ Standard*. This level of compatibility ensures that expansion modules produced by other manufacturers will operate with the ZT 8802. Some of the functions available on expansion modules are as follows (Ziatech product numbers are in parentheses):

- Serial and parallel I/O (zSBX CT32 and zSBX 30)
- Stepper and servo motor controllers
- Analog-to-digital and digital-to-analog converters
- Disk and SCSI controllers
- Modems
- Video controllers
- IEEE 488 controllers (zSBX 20)
- Bar code readers
- Prototyping boards for custom I/O designs (ZT eSBX 70)

Features

The major features of the expansion module interface are listed below.

- Standard interface for expanding I/O capabilities
- Compatible with Intel *iSBX MULTIMODULE Specification*
- Added address lines for custom designs
- Supports both single-wide and double-wide expansion modules

ZT 8802 SPECIFICS

The expansion module interface is supported through connector J8; the pin assignments are given on page B-16.

The expansion module standard defines three address lines and two chip selects. This provides a total of 16 I/O port addresses. To overcome this limitation, the ZT 8802 expansion module adds four address lines. These address lines are connected in the default configuration and can be removed using cuttable traces. The chip select zero signal (MCS0) is valid over the I/O address range FB00 through FB7Fh, and the chip select one signal (MCS1) is valid over the I/O address range FB80 through FBFFh.

The expansion module supports two interrupt request signals for interrupt driven communications. Interrupt request zero (MINTR0) is connected to either IRQ7 (W6) or IRQ4 (W8) of the interrupt controller, and interrupt request 1 (MINTR1) is connected to either IRQ3 (W10) or IRQ2 (W12) of the interrupt controller. The interrupt controller is explained in detail in Chapter 7.

The module present (MPST*) signal is not supported. DMA transfers to the expansion module are not supported.

INSTALLATION

The SBX expansion module is installed on the ZT 8802 as shown in Figure 12-1. The expansion module is mechanically secured to the ZT 8802 at the J8 connector and with the threaded spacer shipped with the module.

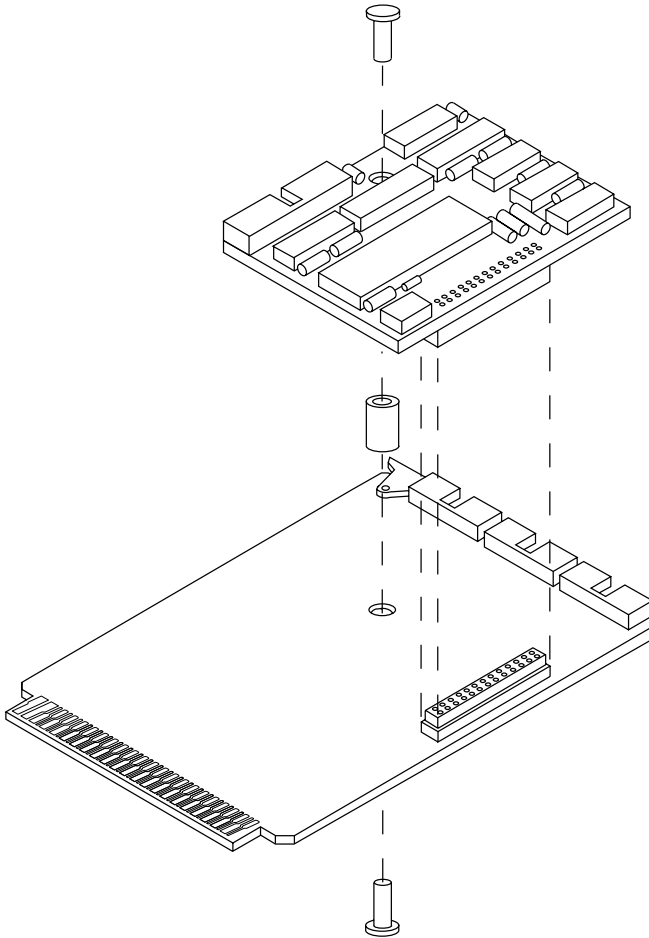


Figure 12-1. SBX Expansion Module Installation.

Chapter 13

PARALLEL I/O ADAPTER (16C49)

Contents	Page
OVERVIEW	13-2
FEATURES OF THE 16C49	13-3
PROCESSOR INTERFACE/CONNECTOR PINOUT	13-4
Control Port	13-5
I/O INTERFACE	13-7
I/O Circuit Operation	13-7
Outputting Data	13-8
Outputting Data With Readback	13-8
Inputting Data	13-8
I/O Circuit Specifications	13-9
Reset Operation	13-9
I/O Port Description	13-10
I/O Port Operation	13-10
Event Sense Port Operation	13-11
Mask Port Operation	13-12
I/O Module Mounting Racks	13-14
ZT 2226 24-Channel I/O Mounting Rack	13-14
ZT 2223 Industrial I/O Adapter Board	13-15
ZT 2225 Industrial I/O Cable Adapter	13-16
Centronics Printer Interface	13-18

OVERVIEW

With a functional block diagram as the basis for discussion (see Figure 13-1 below), this chapter summarizes the features of the 16C49 Parallel Interface Adapter (PIA). Topics discussed include the processor interface/connector pinout and the I/O interface. The chapter concludes with a description of methods for interfacing the ZT 8802 to I/O module mounting racks.

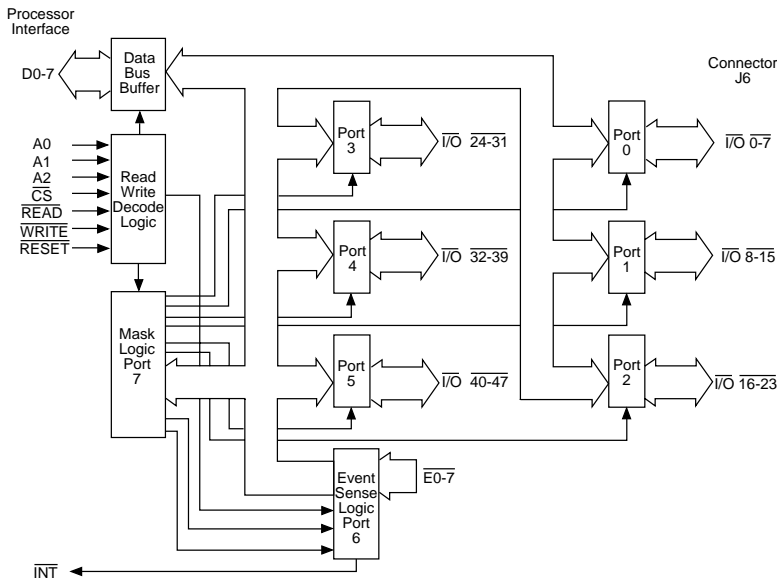


Figure 13-1. 16C49 PIA Block Diagram.

FEATURES OF THE 16C49

The Ziatech 16C49 is an 84-pin high density Parallel Interface Adapter ASIC device for interfacing to digital I/O and Opto 22 or equivalent industrial I/O racks. The 16C49 includes eight ports: six ports consisting of eight I/O points for a total of 48 I/O points per device, a mask port, and an event sense port. The eight ports are I/O mapped from FA00h to FA07h. Each of the 48 I/O points can be used as an output, output with readback, or input.

To avoid false triggering of external devices, the 16C49 will not glitch outputs during power up or power down. Additional 16C49 features include:

- 48 I/O points (low true)
- Open drain outputs
- 50-300 k Ω internal pullups on I/O points
- Mask register to prevent inadvertent writes to I/O points
- Eight positive or negative event sense inputs with interrupt
- Low power, 1 micron CMOS technology
- -40° to +85° Celsius operation

PROCESSOR INTERFACE/CONNECTOR PINOUT

Figure 13-1 on page 13-2 illustrates how the processor interfaces with the 16C49 PIA and connector J6. A simple non-multiplexed bus interface allows data to be written to or read from one of eight ports within the 16C49. Table 13-1 below illustrates the various I/O addressed ports within the 16C49.

Table 13-1
16C49 PIA I/O Ports.

Port #	Port Address	I/O Read Register	I/O Write Register	Connector Pin #s
0000	FA00h	I/O 0-7	I/O 0-7	J6/1-8
0001	FA01h	I/O 8-15	I/O 8-15	J6/10-17
0002	FA02h	I/O 16-23	I/O 16-23	J6/19-26
0003	FA03h	I/O 24-31	I/O 24-31	J6/28-35
0004	FA04h	I/O 32-39	I/O 32-39	J6/37-44
0005	FA05h	I/O 40-47	I/O 40-47	J6/46-53
0006	FA06h	Event Status	Event Clear	
0007	FA07h	Mask Register	Mask Register	

Note: Connector J6 pins 9, 18, 27, 36, and 45 connect to GND. Pins 54, 55, and 56 are connected to +5 V.

The ports within the 16C49 PIA consist of six I/O ports, an event port, and a mask port. Each port is written to or read from by doing an I/O write or read at the board Base Address of the PIA plus the register address specified in Table 13-1. I/O port and mask port operation are covered in "I/O Port Description," page 13-10.

Control Port

The sixth port (at FA05h) is default dedicated for on-board use, but may be configured for off-board I/O at the expense of some on-board features. The items controlled by these eight bits are memory mode selection, LED, watchdog strobe, real-time clock reset, and timer and SBX oscillator disabling. Figure 13-2 on page 13-6 diagrams the bit assignments for these features. Daggers indicate default settings.

Parallel I/O Adapter (16C49)

We recommend you disable interrupts when modifying this register to avoid writing stale data.

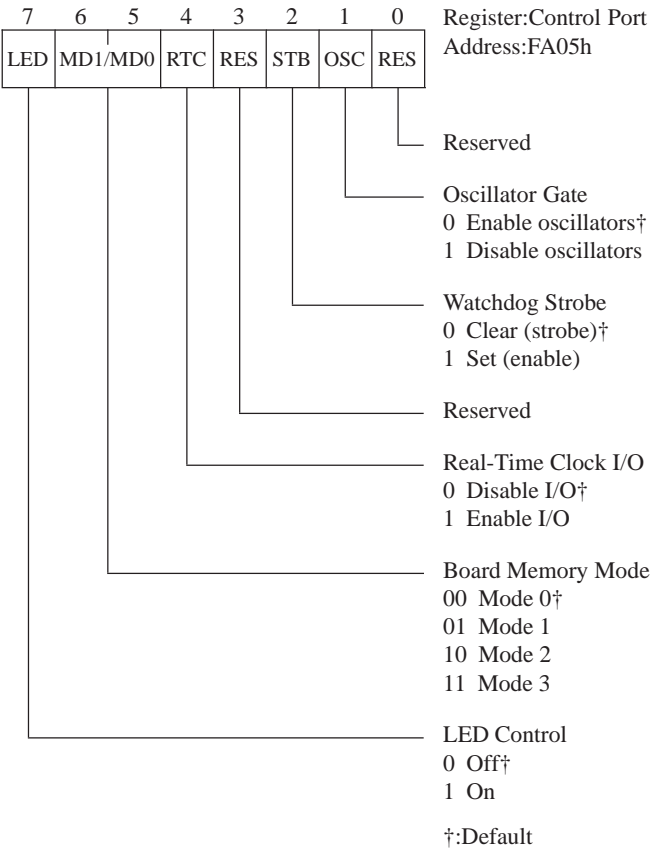


Figure 13–2. Control Port Bit Map.

I/O INTERFACE

I/O Circuit Operation

Each PIA has 48 I/O points in six ports of eight I/O points each. Each I/O point is illustrated in Figure 13-3, "Typical I/O Circuit."

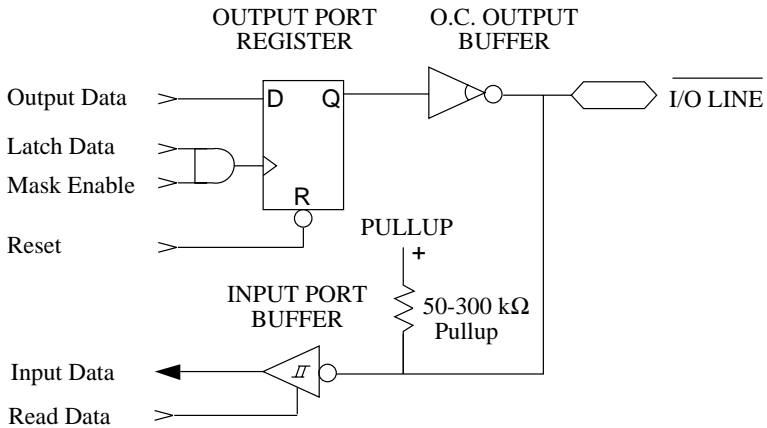


Figure 13-3. Typical I/O Circuit.

Each typical I/O circuit consists of an output register, open collector output buffer, pullup, and input buffer. Input and output operation is described on the following pages.

Outputting Data

Outputting data is accomplished by writing the output data to a given port, thereby causing the latch data signal to capture the output data into the output port register. The register output is buffered by an inverting open collector output buffer before driving the output signal. An integral pullup resistor ensures that a valid high can be measured when the output is not sinking current while in the deasserted or "off" state.

Note: Each output port has a mask enable bit that can prevent inadvertent writes. The mask enable bits are controlled from the mask port and are unmasked to allow writes after power up or reset.

Outputting Data With Readback

Outputting data with readback can be accomplished by outputting data as described above and then reading the input data, thereby causing the read data signal to enable the input port buffer.

Inputting Data

Inputting data is accomplished by reading the input data from a given port, thereby causing the read data signal to input data from the inverting input port buffer. An integral pullup resistor ensures that a valid input is read if the input signal is not connected.

When inputting data the associated circuit must not be used as an output. This allows the output buffer to be inactive, thereby not contending with the input signal. The output circuits are inactive when a logical 0 is output, after reset, or after power up.

When using a port configured with some output and some input circuits, care must be taken to ensure that any circuits used as inputs are always written with a logical 0.

I/O Circuit Specifications

Table 13-2 lists the electrical specifications of each I/O point.

Table 13-2
I/O Point Specifications.

Parameter	Specification
Output Sink Current (Iol)	12 mA min.
Output Low Voltage (Vol at Iol)	.4 V max.
Internal Pullup	50 k Ω - 300 k Ω

Reset Operation

Each I/O circuit on the 16C49 PIA is reset automatically by the reset input. The reset input is connected to a precision nonglitching reset circuit on the ZT 8802 to prevent the output circuits from glitching on power up or power down. The reset circuit is active when the supply voltage is within 0-4.75 V.

After SYSRESET* is asserted, the mask register is initialized to enable (unmasked) writing to the I/O registers.

I/O Port Description

I/O ports are used to communicate with the 16C49. Six I/O ports are used to output and input data. An event sense port can be used to determine input transitions on I/O points, and a mask register is used to prevent inadvertent writes to the outputs. Each port is described on the following pages.

I/O Port Operation

The six I/O ports implement eight I/O points each. Data bit D0 corresponds to the least significant I/O point in each port. Table 13-3 lists the port and I/O point assignments for outputs and inputs.

Table 13-3
I/O Point Write/Read Ports.

Port	D7	D6	D5	D4	D3	D2	D1	D0
0	$\overline{\text{I/O } 7}$	$\overline{\text{I/O } 6}$	$\overline{\text{I/O } 5}$	$\overline{\text{I/O } 4}$	$\overline{\text{I/O } 3}$	$\overline{\text{I/O } 2}$	$\overline{\text{I/O } 1}$	$\overline{\text{I/O } 0}$
1	$\overline{\text{I/O } 15}$	$\overline{\text{I/O } 14}$	$\overline{\text{I/O } 13}$	$\overline{\text{I/O } 12}$	$\overline{\text{I/O } 11}$	$\overline{\text{I/O } 10}$	$\overline{\text{I/O } 9}$	$\overline{\text{I/O } 8}$
2	$\overline{\text{I/O } 23}$	$\overline{\text{I/O } 22}$	$\overline{\text{I/O } 21}$	$\overline{\text{I/O } 20}$	$\overline{\text{I/O } 19}$	$\overline{\text{I/O } 18}$	$\overline{\text{I/O } 17}$	$\overline{\text{I/O } 16}$
3	$\overline{\text{I/O } 31}$	$\overline{\text{I/O } 30}$	$\overline{\text{I/O } 29}$	$\overline{\text{I/O } 28}$	$\overline{\text{I/O } 27}$	$\overline{\text{I/O } 26}$	$\overline{\text{I/O } 25}$	$\overline{\text{I/O } 24}$
4	$\overline{\text{I/O } 39}$	$\overline{\text{I/O } 38}$	$\overline{\text{I/O } 37}$	$\overline{\text{I/O } 36}$	$\overline{\text{I/O } 35}$	$\overline{\text{I/O } 34}$	$\overline{\text{I/O } 33}$	$\overline{\text{I/O } 32}$
5	$\overline{\text{I/O } 47}$	$\overline{\text{I/O } 46}$	$\overline{\text{I/O } 45}$	$\overline{\text{I/O } 44}$	$\overline{\text{I/O } 43}$	$\overline{\text{I/O } 42}$	$\overline{\text{I/O } 41}$	$\overline{\text{I/O } 40}$

Writing the specified port with the corresponding data bit set (1) causes the I/O output to become active (low). Writing a port with the bit reset (0) causes the I/O output to become inactive (high).

Reading the specified port returns the status of the input point. A high bit (1) indicates the presence of a low true input (0). A low bit (0) indicates a high input (1) is present.

Event Sense Port Operation

The event sense port is used to optionally determine input transitions on I/O points (polarity is selected by the mask port). Event status of the eight event inputs (E0-7) is read and event sense status clearing is done via the event sense port (see Table 13-4).

Table 13-4
Event Sense Port.

Port	D7	D6	D5	D4	D3	D2	D1	D0
FA06h J6 pin	E7 8	E6 7	E5 6	E4 5	E3 4	E2 3	E1 2	E0 1

When reading the event sense port, each bit being set to a logical 1 indicates an event on that input has occurred.

When writing the event sense port, each data bit written with a logical 0 clears its corresponding event sense flip-flop. Each data bit of the event sense port must be written with a logical 1 to re-enable the corresponding event sense input after it is cleared or after power up or reset.

Parallel I/O Adapter (16C49)

The event sense inputs are connected to the same signals as Port 0 (FA00h), as shown in Figure 13-4 below.

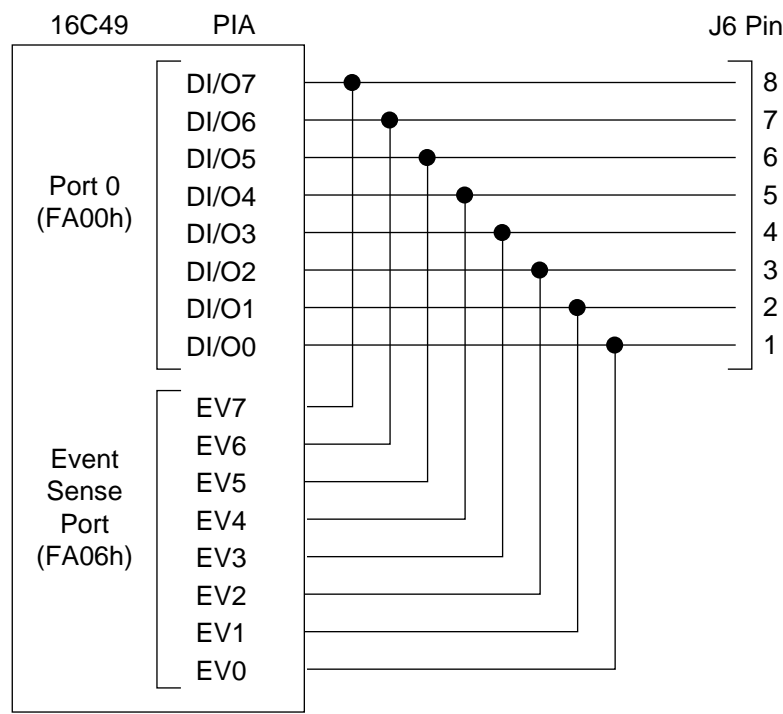


Figure 13-4. Event Sense Inputs.

Mask Port Operation

The mask register is used to prevent writing the output ports unless enabled. Power-up default has the register unmasked to allow writes to the output ports. Writing the mask register bits D0-5 with a one (1) masks I/O ports 0-5 respectively. Reading the mask port is allowed to determine which ports are enabled.

Table 13-5 illustrates the data bit and mask port relationships.

Table 13-5
Mask Port.

Port	D7	D6	D5	D4	D3	D2	D1	D0
FA07h (write)	E7-4	E3-0	Port 5	Port 4	Port 3	Port 2	Port 1	Port 0
FA07h (read)	Int	0	Port 5	Port 4	Port 3	Port 2	Port 1	Port 0

The upper two bits in the mask register are used in conjunction with the event sense port. When writing, the upper two bits of the mask register select the polarity sensed by event sense inputs E0-E7, which are connected to I/O points $\overline{I/O}$ $\overline{0-7}$ on the ZT 8802. Bits 7 and 6 determine E7-4 and E3-0, respectively. Writing a logical 0 (power-up default) senses negative events (edges), while writing a logical 1 senses positive events.

When reading the mask register, the most significant bit (D7) returns the interrupt signal status on the interrupt output pin of the 16C49. A logical 1 means the interrupt is asserted.

I/O Module Mounting Racks

The parallel I/O is often used to manage one or two industrial I/O module mounting racks such as those offered by Ziatech (ZT 2226) or Opto 22. This section discusses several methods for interfacing the ZT 8802 to I/O module mounting racks.

Note: Be sure to read Appendix C, *PIA System Setup Considerations*, before interfacing I/O module mounting racks to the 16C49 PIA, since the CMOS technology utilized in the PIA can exhibit latchup characteristics under certain conditions. Appendix C illustrates precautions you should take to prevent latchup.

ZT 2226 24-Channel I/O Mounting Rack

The ZT 2226 is a 24-channel I/O module mounting rack that connects directly to the ZT 8802. It holds up to 24 I/O modules and can be panel mounted using its integral mounting hardware. This setup can reduce the size of the control system and free up valuable enclosure space compared to other I/O interface setups.

A 56-pin header on the ZT 2226 provides a direct interface to the parallel port header (J6) on the ZT 8802. The 56-pin header is pin-for-pin compatible, accepts the same cable connector, and has the same electrical specifications as the ZT 8802 header. A typical interface cable is a standard 56-conductor ribbon cable such as the ZT 90089 (40 inches in length). Power for the ZT 2226 can be supplied across the ribbon cable or by a set of terminals that connect to a remote power supply. The 56-pin interface eliminates custom cables, adapter cards, and I/O interface cards.

You can daisy-chain a second ZT 2226 or other 24-channel I/O rack to the first ZT 2226 to provide a total of 48 I/O module positions. To accommodate this, the 48 I/O signals are divided into two sets of 24. One set is connected to the first ZT 2226. The second set is connected to the 50-pin daisy-chain header. The daisy-chain header is connected to a second I/O rack using a 50-conductor ribbon cable such as the ZT 90022 (9.5 inches), ZT 90072 (10 feet), or ZT 90137 (2 feet).

Combine the ZT 8802 and the ZT 2226 to create a compact control system with 24 I/O module positions that can be configured specifically for your application. I/O modules are available from Opto 22 and Grayhill and range in function from DC output types to Quadrature input types. I/O modules are also available with extended operating temperature capabilities. The ZT 2226 provides large 10 AWG terminals to connect field wiring to the I/O modules.

ZT 2223 Industrial I/O Adapter Board

In situations where it is impossible or undesirable to use the ZT 2226, Ziotech recommends the ZT 2223 Industrial I/O Adapter Board. The ZT 2223 converts the ZT 8802's 56-pin parallel I/O interface to two 50-pin, I/O mounting rack compatible connectors with alternating signal and ground lines.

The ZT 2223 is intended primarily for connecting to I/O module mounting racks that do not accept Opto 22 Generation 4® (G4) miniature modules. Manufacturers include Grayhill, Potter & Brumfield, Gordos, Crydom, Analog Devices, and others. The ZT 2223 is not designed to mount to racks using a card-edge connector.

The ZT 2223 interfaces with the ZT 8802's parallel I/O connector (J6) via a standard 56-conductor ribbon cable such as the ZT 90089 (40 inches). This interface eliminates custom cables and I/O interface cards. The ZT 2223's 56-pin header is pin-for-pin compatible, accepts the same cable connector, and has the same electrical specifications as the ZT 8802 header.

The ZT 2223 then plugs directly onto the first I/O mounting rack using a downward-facing 50-pin connector and is secured to the mounting plate with standoffs. A second 50-pin header provides a daisy-chain cable interface for a second I/O mounting rack. Use any of the 50-conductor cables listed on page 13-14.

Power for the ZT 2223 is jumper selectable from the 56-conductor ribbon cable.

ZT 2225 Industrial I/O Cable Adapter

As a third option, the ZT 2225 Industrial I/O Cable Adapter can be used to interface the ZT 8802 to one or two I/O module mounting racks. The ZT 2225 plugs directly onto the ZT 8802 J6 parallel I/O connector in a piggyback fashion and accepts one or two ZT 90072 cables (50-pin both ends) or ZT 90021 cables (50-pin header at one end, card-edge connector at the other). Each cable interfaces the ZT 2225 directly to an I/O module mounting rack. The ZT 2225 can be mounted externally to the card cage if an empty card cage slot is not available.

Table 13-6 shows the relationship between the I/O port address of each ZT 8802 parallel I/O bit and the associated pin on the ZT 2225.

As a word of caution, note that the connector type and pin assignments of different I/O module mounting racks vary significantly. Always compare the ZT 2225 pin assignments given in Table 13-6 to those of the I/O module mounting rack to ensure compatibility.

Note also that the ZT 90021 and ZT 90072 digital I/O cables mentioned above supply +5 V power on pin 49. *When ribbon cable power is not desired, be sure the I/O rack ribbon cable power jumper is removed* or use a ZT 2226 I/O mounting rack.

Table 13-6
ZT 8802 J6 Connection to ZT 2225.

J6 Pin	Signal	Port Address [hex]	J6 Pin	Signal	Port Address [hex]
1	IO00/E0	FA00 bit 0	31	IO27	FA03 bit 3
2	IO01/E1	FA00 bit 1	32	IO28	FA03 bit 4
3	IO02/E2	FA00 bit 2	33	IO29	FA03 bit 5
4	IO03/E3	FA00 bit 3	34	IO30	FA03 bit 6
5	IO04/E4	FA00 bit 4	35	IO31	FA03 bit 7
6	IO05/E5	FA00 bit 5	36	GND	--
7	IO06/E6	FA00 bit 6	37	IO32	FA04 bit 0
8	IO07/E7	FA00 bit 7	38	IO33	FA04 bit 1
9	GND	--	39	IO34	FA04 bit 2
10	IO08	FA01 bit 0	40	IO35	FA04 bit 3
11	IO09	FA01 bit 1	41	IO36	FA04 bit 4
12	IO10	FA01 bit 2	42	IO37	FA04 bit 5
13	IO11	FA01 bit 3	43	IO38	FA04 bit 6
14	IO12	FA01 bit 4	44	IO39	FA04 bit 7
15	IO13	FA01 bit 5	45	GND	--
16	IO14	FA01 bit 6	46	IO40	FA05 bit 0 (DC/DC)
17	IO15	FA01 bit 7	47	IO41	FA05 bit 1 (OSC)
18	GND	--	48	IO42	FA05 bit 2 (WDSTB)
19	IO16	FA02 bit 0	49	IO43	FA05 bit 3 (Reserved)
20	IO17	FA02 bit 1	50	IO44	FA05 bit 4 (RTC)
21	IO18	FA02 bit 2	51	IO45	FA05 bit 5 (MD0)
22	IO19	FA02 bit 3	52	IO46	FA05 bit 6 (MD1)
23	IO20	FA02 bit 4	53	IO47	FA05 bit 7 (LED)
24	IO21	FA02 bit 5	54	+5V	-- 1A fused
25	IO22	FA02 bit 6	55	+5V	-- 1A fused
26	IO23	FA02 bit 7	56	+5V	-- 1A fused
27	GND	--			
28	IO24	FA03 bit 0			
29	IO25	FA03 bit 1			
30	IO26	FA03 bit 2			

Centronics Printer Interface

Two methods are used to interface the ZT 8802 to a standard IBM printer cable. The first method uses the ZT 90156, an 8" (20.3 cm) printer interface cable with a 56-pin transition connector on one end and a 25-pin female D-shell connector on the other. Figure B-7 on page B-20 illustrates the ZT 90156 cable and its pin assignments. If you prefer to make a custom cable, we recommend a length of three feet or less.

The second method by which to interface the ZT 8802 to a Centronics printer is the ZT 2225 Industrial I/O Cable Adapter mentioned earlier. It plugs directly onto the ZT 8802 J6 parallel I/O connector and accepts the ZT 90028 cable, which then interfaces J4 of the ZT 2225 to a standard IBM printer cable. Note that the ZT 2225 can be mounted externally to the card cage if an empty card cage slot is not available.

REAL-TIME CLOCK/CALENDAR (DS 1202)

Contents	Page
OVERVIEW	14-2
ZT 8802 SPECIFICS	14-2
FUNCTIONAL DESCRIPTION	14-3
OPERATION	14-3
COMMAND BYTE	14-5
Byte Mode	14-5
Burst Mode	14-6
Byte Mode Versus Burst Mode	14-7
Data Input/Output	14-7
Clock/Calendar	14-8
Clock Halt Flag	14-10
AM-PM/12-24 Hour Mode	14-10
Write-Protect Register	14-10
Clock/Calendar Burst Mode	14-10

OVERVIEW

The function of the real-time clock is to preserve the time of day through the host board's power cycle. During the boot sequence, an operating system reads the real-time clock (RTC) and maintains the time of day for system use. In addition to the time of day, the RTC is responsible for keeping the day, date, and year. Some RTCs also keep track of such things as leap years and end of month rollovers for months shorter than 31 days.

In the case of Ziatech DOS, the system time maintained while power is on is independent of the RTC. One of the timer/counters (Timer 0) generates a periodic interrupt every 54.93 milliseconds for the BIOS to update a memory area that is then used by applications for time or date information. Note that under DOS, when the "time" command is typed at the command prompt, the BIOS does not read the RTC, it reads its own real time, which the timer/counter is updating. The RTC is used to provide the initial time only at boot up.

ZT 8802 SPECIFICS

The ZT 8802 uses the Dallas Semiconductor DS 1202 Serial Timekeeper Chip. Communication is performed with the V40 microprocessor via I/O port FA80h, bit 0. Reads and writes to this bit allow the serial data of the RTC to be communicated. Through this interface, command and data bytes are read and written. This port is redundantly mapped in the I/O space from FA80h through FAFFh. In addition, a Control Port bit (FA05h, bit 4) is used to enable and disable communication with the DS 1202. Note that W45 must be installed for the RTC to be interfaced.

FUNCTIONAL DESCRIPTION

The DS 1202 contains a real-time clock and calendar and 24 bytes of static RAM. The real-time clock subsystem provides seconds, minutes, and hours. The calendar provides day, date, month, and year information. The end of the month is automatically adjusted for months with less than 31 days. In addition, leap year compatibility is provided. The clock can be configured for either 12 or 24 hour mode. In 12 hour mode, an AM/PM indicator is provided. Data may be transferred to and from the RTC one byte at a time or in a burst of up to 24 bytes. The DS 1202 is designed to operate on very low power, using a maximum of 1.2 mA in operation and 300 nA maximum in data retention mode (battery backup).

OPERATION

The real-time clock/calendar (DS 1202) is read and written from FA80h bit 0. The RTC must be enabled for I/O by writing a logical 1 to bit 4 of the Control Port. Be careful not to alter other bits in the Control Port. In addition, the SBX port (I/O FB00h-FBFFh) must not be read or written while the RTC I/O pin is enabled. The following sequence must be followed for correct RTC operation:

1. Clear interrupts.
2. Enable the RTC I/O pin (set FA05h, bit 4).
3. Re-enable interrupts (optional).
4. Perform RTC I/O (via FA80h, bit 0).
5. Clear interrupts (if enabled in step 3).
6. Disable the RTC I/O pin (clear FA05h, bit 4).
7. Re-enable interrupts.

For example, to set bit 4, the following assembly language code would be used:

```
cli                ; clear interrupts
mov dx, 0Fa05h    ; Control Port
in  al, dx        ; read it
or  al, 010h      ; enable RTC I/O
out dx, al        ;
sti
```

To initiate any data transfer to the RTC, an 8-bit command code is written via FA80h bit 0 after RTC I/O is enabled (via the Control Port FA05h, bit 4). This command byte specifies which of the 32 bytes are accessed, whether a read or write cycle is to be performed, and whether a byte or burst transfer will take place. Following the command byte, subsequent cycles then either read or write data through bit 0 of FA80h. There are 24 bytes (192 bits) of RAM, and the clock/calendar is implemented in 8 bytes (64 bits). After data is read or written, it is important that the RTC I/O pin be disabled via the Control Port. The following code accomplishes this:

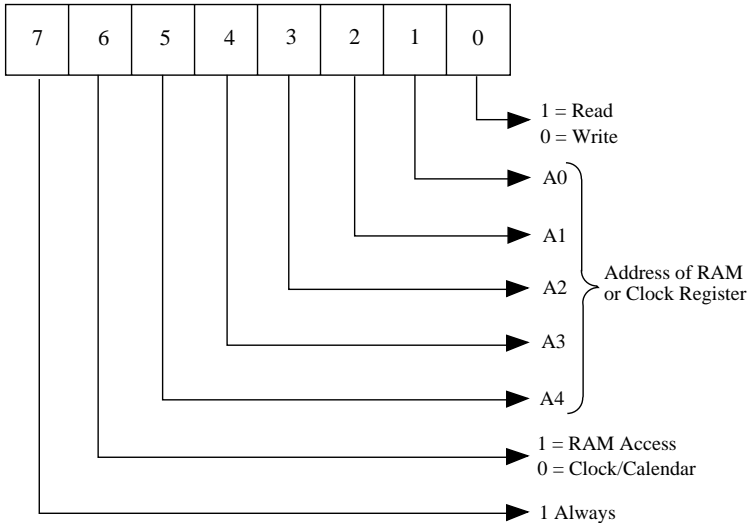
```
cli                ; clear interrupts
mov dx, 0Fa05h    ; Control Port
in  al, dx        ; read it
and al, 0EFh      ; disable RTC I/O
out dx, al        ;
sti
```

This step must be undertaken before a new location (in byte mode) or a new burst access is undertaken.

COMMAND BYTE

Byte Mode

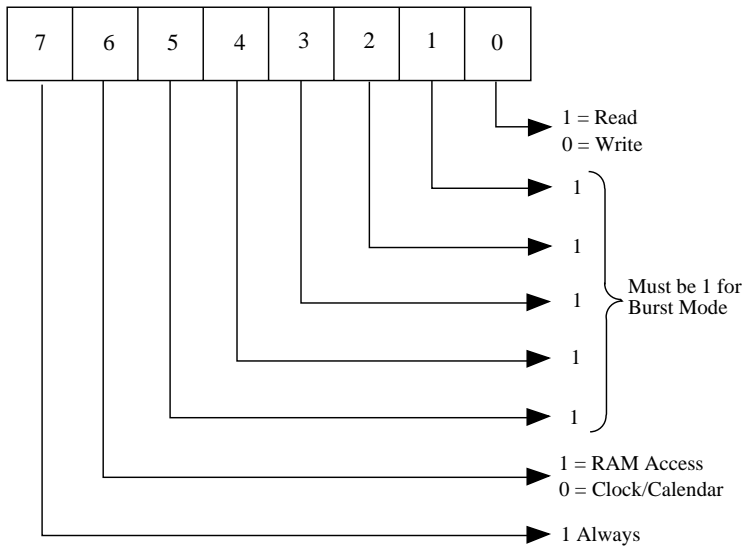
The command byte is shown in Figure 14-1. Each transfer is initiated by a command byte. The Most Significant Byte (bit 7) must be a logical 1. If it is a logical 0, further action is terminated. Bit 6 specifies whether the clock/calendar or the general purpose RAM is being accessed. Ziatech operating systems do not use the RAM, so it is available for application use. If this bit is a logical 1, then RAM is to be accessed. If it is a logical 0, then the clock/calendar will be accessed. Bits 1-5 specify which of the 24 RAM locations will be accessed (bytes 0-23). Bytes 24-31 are not provided. Bit 0 specifies a write operation if logical 0 or a read operation if logical 1. The command byte is written starting with bit 0.



*Figure 14-1. Command Byte Bit Format (Byte Mode).
Byte reads/writes.*

Burst Mode

Both the clock/calendar and RAM may be read/written in a software burst by initiating a burst mode command. The burst mode command is the same as a byte command except the address bits (bits 1-5) are all set to logical 1 (see Figure 14-2 below). After the burst mode command has been programmed, all RAM or clock/calendar bytes are read or written as a stream of serial I/O without having to send a command for each location. The 24 bytes of RAM take 192 reads or writes via FA80h to complete the sequence. The eight bytes of clock/calendar data take 64 accesses to FA80h to complete the sequence.



*Figure 14–2. Command Byte Bit Format (Burst Mode).
Burst reads/writes.*

Byte Mode Versus Burst Mode

Byte mode is useful when only a portion of the RAM or clock/calendar needs to be accessed, such as the seconds register. For a few bytes only, this mode is more efficient because it does not have the additional overhead of reading and writing the remaining unused bytes as in a burst transfer. A burst transfer is more efficient for applications that utilize most or all of the RAM or clock/calendar. For reference, Ziatech's DOS interfaces to the RTC in byte mode. Depending upon how the interface code is structured, you must decide which mode is more efficient. Remember, we recommend that you disable interrupts during RTC accesses, which may preclude burst mode operation.

Data Input/Output

Following the command byte, the data for the type of command being performed is read or written in either a byte or burst. Again the RTC Data Port (FA80h, bit 0) is used to serially read or write each data bit of each byte. Additional reads or writes beyond that for which the command port is programmed are ignored. Following the data transfer, the RTC I/O pin must be disabled by clearing bit 4 of the Control Port to allow new commands to be written. Data is read or written starting with bit 0 of port FA80h.

An example of code reading from byte 5 of the RAM is as follows:

```
enable_rtc_io:
    cli                ; clear interrupts
    mov dx, 0Fa05h     ; Control Port
    in  al, dx         ; read it
    or  al, 010h       ; enable RTC I/O
    out dx, al         ;

command_byte:
    mov al, 11001011b  ; command byte CBh
    mov dx, 0FA80h     ; RTC address
    mov cx, 8          ; 8 data bits

out_data:
    out dx, al         ;
    ror al, 1          ; get next bit
    loop out_data      ;
    mov cx, 8

get_data:
    in  al, dx
    ror ax, 1          ; shift into ah
    loop get_data      ; get next bit
    xchg al, ah        ; assume es and di setup and
    stosb              ; store at ES:[di]

disable_rtc_io:
    mov dx, 0Fa05h     ; Control Port
    in  al, dx         ; read it
    and al, 0EFh       ; disable RTC I/O
    out dx, al         ;
    sti
```

Note that the above code disables interrupts throughout the reading of the RTC. This might not be feasible in some systems due to interrupt latency issues. At a minimum, the Control Port read/write sequence should be protected from interrupts. If interrupts are enabled during the RTC data sequence, be careful during system design to not have interrupt service routines also modify the RTC or read the SBX port.

Clock/Calendar

The clock/calendar is contained in eight write/read registers. Each byte is accessible individually or as part of a burst read or write. In burst mode, 64 reads or writes are needed to read all 8 bytes. The data is in Binary Coded Decimal (BCD) format. The registers are shown in Figure 14-3.

Command Bytes/Definition

Register	Function	Command Address (HEX)	Write=W Read=R	Range Data (BCD)	Register Definition							
					7	6	5	4	3	2	1	0
0	Seconds	80 81	W R	00-59	Ch	10 Sec				Sec		
1	Minutes	82 83	W R	00-59	0	10 Min				Min		
2	12 Hrs. 24 Hrs.	84 85	W R	01-12 00-23	12\24	0	AP	HR	HR	Hour		
3	Date	86 87	W R	01-31	0	0	10 Date			Date		
4	Month	88 89	W R	01-12	0	0	0	10M	Month			
5	Day	8A 8B	W R	01-07	0	0	0	0	Day			
6	Year	8C 8D	W R	00-99	10 Year				Year			
7	Write Protect	8E 8F	W R	00-80	WP	Always Zero						

31	Clock Burst	BE	W
		BF	R
0 ⋮ 23	RAM 0 ⋮ RAM 23	C0	W
		C1	R
		⋮	⋮
31	RAM Burst	EE	W
		FE	W
		FF	R

Figure 14-3. Clock/Calendar Registers.

Clock Halt Flag

Bit 7 of the seconds register is used to halt the real-time clock for extreme low-power usage. When this bit is set to 1, the DS 1202 is placed in low-power mode, drawing no more than 100 nA. In ZT 8802 applications, low-power mode is not necessary. The battery used has a shelf life of 10 years, which is equivalent to a usage of 11 μ A. Assuming neither of the static RAM sockets is being battery backed, the RTC pulls only 1 μ A maximum, which means the shelf life of the battery is the limiting factor.

AM-PM/12-24 Hour Mode

Bit 7 of the hours register is defined as the 12- or 24-hour mode select bit. When set high, the 12-hour mode is selected. In the 12-hour mode, bit 5 is the AM/PM bit. A logic high indicates PM. In 24-hour mode, bit 5 is the second 10 hour bit (20-23 hours).

Write-Protect Register

Bit 7 of the write-protect register is the write-protect bit. The first seven bits (bits 0-6) are forced to 0 when read. Before any write operation to the clock or RAM, bit 7 must be 0. When high, the write protect bit prevents a write operation to any other register.

Clock/Calendar Burst Mode

The clock/calendar command byte specifies burst mode operation if bits 1-5 are set high. In this mode all 8 bytes are read or written starting with bit 0 of register 0. Data is read/written via I/O port FA80h, bit 0.

Appendix A

JUMPER CONFIGURATIONS

Contents	Page
OVERVIEW	A-1
JUMPER OPTIONS	A-3

OVERVIEW

The ZT 8802 includes jumper options that allow you to tailor the board's operation to the requirements of specific applications.

Table A-1 on page A-2 divides the jumpers into functional groups and lists page numbers where descriptions of the jumpers can be found.

Table A-2 lists the jumpers in numerical order and provides a description of each. A dagger (†) in Table A-2 indicates a standard default jumper configuration, which is for DOS operation.

Figure A-4 on page A-16 illustrates jumper pin locations. You may wish to document your custom jumper configuration using this figure.

Figures A-5 and A-7 on pages A-17 and A-19 show the default jumper configurations for Ziatech DOS and STD ROM, respectively. Figure A-6 on page A-18 shows the jumper configuration for DOS with 512K ROM and 1 Mbyte RAM.

Jumper Configurations

Table A-1
Jumper Cross Reference.

Function	Jumper #	Pages
Battery Backup	W1, 28-29, 32-33	A-3
Control Port	W4, 22-25, 45-46	A-10
Miscellaneous:		
Digital Ground & AUX GND	W60	A-13
MEMEX (BHE*)	W52	A-12
Serial Controller Freq.	W26-27	A-12
RAM Configuration	W19-20, 30-31, 34-35	A-4
ROM Configuration	W21, 40-44, 38-39	A-5, A-6
STD Bus Interrupts	W5-18, 47-51, 53-59	A-7
Timer Clock Selection	W2-3	A-13
Factory Options	W36-37, CT1-2, CT13-17, CT19-20†	A-14, A-15

† CT numbers refer to cuttable traces located on the solder side of the board.

JUMPER OPTIONS

Table A-2
Jumper Descriptions.

JUMPER #	DESCRIPTION			
W1, W28-29,	Battery Backup Device Selection — determines if the local RAM devices, RAM LOW and RAM HIGH, are powered by the battery when the system power is turned off. Note that this operation is valid only if the optional battery is present. The RAM devices should be designed for low power operation (less than 15 μ A data retention current). W1A is used to supply battery power to the RAM. With power off, W1B may be installed to erase the battery-backed RAM and real-time clock contents. The RAM HIGH is socket 4F. The RAM LOW is socket 3F. In a DOS system, both sockets should be battery backed as shown. Default installs W1A and removes W1B.			
W32-33				
W32	W33	W28	W29	Battery-Backed Devices
OUT†	IN†	X	X	RAM HIGH battery backed
IN	OUT	X	X	RAM HIGH not battery backed
X	X	OUT†	IN†	RAM LOW battery backed
X	X	IN	OUT	RAM LOW not battery backed

† Factory default (DOS) configuration.

Jumper Configurations

Table A-2
Jumper Descriptions (continued).

JUMPER #		DESCRIPTION
W19-20, W30-31,		RAM Size Configuration. These jumpers allow configuration of the RAM installed. RAM HIGH is socket 4F, and RAM LOW is socket 3F.
W34-35		

W35	W34	W31	W30	W20	W19	RAM LOW	RAM HIGH
OUT	IN	OUT	IN	IN	IN	512K	512K
IN†	OUT†	IN†	OUT†	IN†	OUT†	128K	128K
IN	OUT	OUT	IN	OUT	IN	512K	—
IN	OUT	IN	OUT	OUT	OUT	128K or smaller	—

† Factory default (DOS) configuration.

Table A-2
Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W21, W40-44	Flash Memory/ROM Socket Configuration. These jumpers select the function for several pins at socket 5F. 5 V and 12 V Flash may be used, as well as PROM/EPROM. W21 controls the address space that the memory decoder allocates for the ROM/Flash.
Jumper	Description
Addr. Size	
W21 IN	Selects 512K ROM/Flash‡
W21 OUT†	Selects 256K or smaller ROM/Flash
Pin 1 Select.	
W42	Supplies pin 1 of ROM/Flash socket with A18 (5 V 512K Flash only)‡
W43†	Supplies pin 1 of ROM/Flash socket with Vcc (write protects Flash)
W44	Supplies pin 1 of ROM/Flash socket with +12 V from backplane (allows Flash programming)
Pin 31 Select.	
W40	Supplies A18 for 512K ROM socket pin 31
W41†	Supplies write signal to Flash pin 31 (Flash only)

† Factory default (DOS) configuration.

‡ The maximum +12 V Flash size is 256K (W21 OUT, W42 OUT).

Table A-2
Jumper Descriptions (continued).

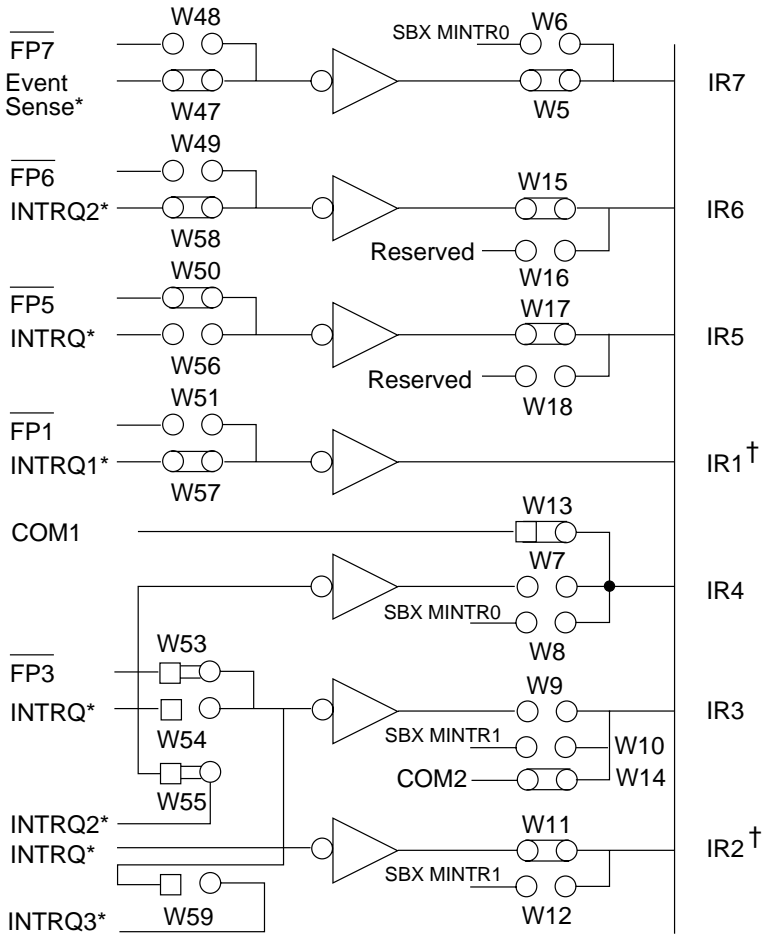
JUMPER #		DESCRIPTION
W38-39		28-Pin/32-Pin Device Selection. It is possible to load ROM devices that are smaller than 128K onto the ZT 8802. For cost-sensitive systems, this may be an effective way to reduce costs. These devices are redundantly mapped within the top 128K of memory. STD ROM is shipped in 32K EPROM.
W39	W38	Device Selection
OUT†	IN†	32-pin ROM/Flash
IN	OUT	28-pin ROM (for example, 32K ROM)

† Factory default (DOS) configuration.

Table A-2
Jumper Descriptions (continued).

JUMPER #	DESCRIPTION
W5-18, W47-51, W53-59	STD and Frontplane Interrupt Selection. These jumpers allow configuration of the interrupt subsystem. Interrupts may be driven by the backplane, frontplane, or expansion module. The defaults are shown in Figure A-1 on page A-8. Note that the IR4 and IR3 levels may be alternately jumpered by using right angle jumpers at W54 and W55; see Figure A-2 on page A-9.

Jumper Configurations



† IR1 and IR2 have other options within the V40. Refer to the OPCN register in Chapter 5 for details.

Figure A-1. ZT 8802 Interrupt Jumper Selection.

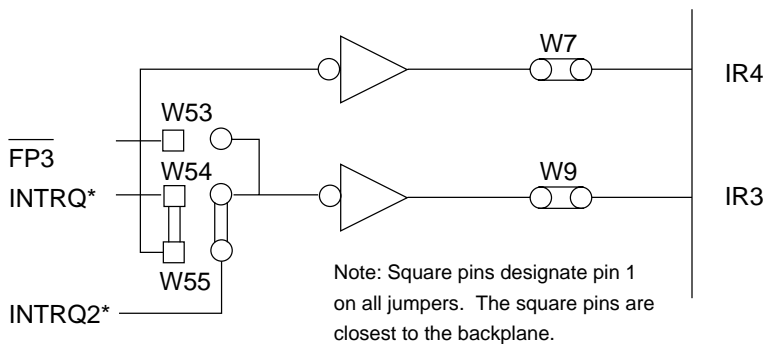


Figure A-2. Alternate IR4/IR3 Jumper Selection.

Table A-2
Jumper Descriptions (continued).

JUMPER #	DESCRIPTION	
W4, W22-25,	Control Port Configuration. The Control Port (FA05h) is part of the 16C49 PIA interface. These jumpers allow software control over several features of the ZT 8802.	
W45-46		
Jumper	Function	
W45	IN†	Controls real-time clock I/O pin enable. When installed, writing logical 1 to bit 4 of the Control Port enables the I/O pin on D0 of the data bus.
	OUT	Disables RTC I/O.
W46	IN	Allows watchdog strobe signal to be driven by bit 2 of the Control Port.
	OUT†	Disables watchdog capability.
W4	IN	Allows Control Port bit 1 to three-state SBX and system timer oscillators.
	OUT†	Enables oscillators.
W22,24	IN†	Allow Control Port bits 5 and 6 (MD0 and MD1 respectively) to drive memory map mode bits. W23 and W25 must be removed.
	OUT	Allow W23 and W25 to be installed.
W23,25	IN	Allow hardwiring of memory map mode bits if W22, W24 removed.
	OUT†	Allow W22 and W24 to be installed.

† Factory default (DOS) configuration; STD ROM systems have W45-46 OUT, W4 OUT, W22-25 OUT.

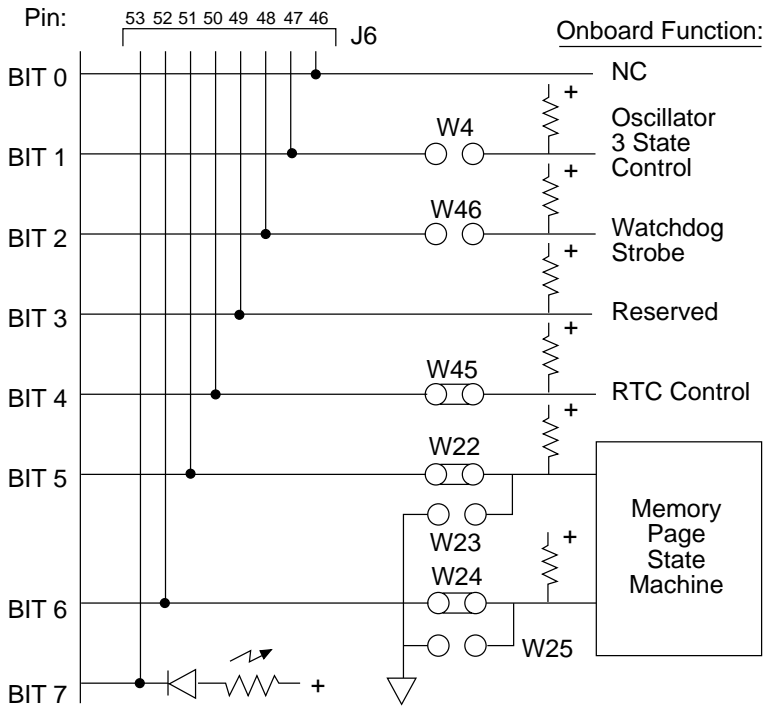


Figure A-3. Control Port Configuration.

Jumper Configurations

Table A-2
Jumper Descriptions (continued).

JUMPER #		DESCRIPTION
W26-27		Serial Controller Frequency. The serial controller can be driven by either an 8 MHz or a 1.8432 MHz oscillator. <i>Do not install both jumpers at the same time.</i>
W26	W27	Description
IN†	OUT†	Select 1.8432 MHz (DOS default)
OUT	IN	Select 8.000 MHz (custom)
W52		MEMEX (BHE*).
Jumper		Description
W52 IN		Forces MEMEX to be driven low (0 V) for all memory boards (some boards need this).
W52 OUT†		MEMEX is pulled up high. This signal is the STD 32 BHE* signal and should not be driven low when using the ZT 8802 with these peripherals.

† Factory default (DOS) configuration.

Table A-2
Jumper Descriptions (continued).

JUMPER #		DESCRIPTION
W60		Digital Ground and AUX GND.
Jumper		Description
W60 IN†		Electrically connects the AUX ground pins on the backplane to digital ground.
W60 OUT		Allows digital ground and AUX GND to be at different potentials.
W2-3		Timer Clock Selection. The V40 timer/counters have two sources for the TCLK input. For DOS systems, this clock is driven by a 1.19318 MHz frequency for DOS compatibility. Alternately, this clock input can be driven from frontplane connector J2, pin 2.
W3	W2	Timer/Counter Clock Frequency
OUT†	IN†	1.19318 MHz
IN	OUT	Selected by J2, pin 2

† Factory default (DOS) configuration.

Table A-2
Jumper Descriptions (continued).

TRACE #		DESCRIPTION
CT1-CT2		Factory Option: Watchdog Strobe Interval. These factory installed 0 Ω resistors allow configuration of the watchdog timer interval, which is selectable from one of three periods. This time interval defines the amount of time that an application can wait to strobe the watchdog timer before a system reset is generated. The minimum time of each range is to be used for software planning. The maximum of each range defines how long the application must wait for a reset if the integrity of the system is lost. <i>Note that these are factory options only.</i>
CT2	CT1	Description
IN†	OUT†	Watchdog interval 400-2000 ms, 1200 ms typical
OUT	OUT	Watchdog interval 250-1000 ms, 600 ms typical
OUT	IN	Watchdog interval 50-250 ms, 150 ms typical
IN	IN	NOT ALLOWED! Shorts Vcc to ground

† Factory default. (Factory option only.)

Table A-2
Jumper Descriptions (continued).

JUMPER #		DESCRIPTION
W36-37, CT13-17, CT19-20		Factory Option: RAM Socket Conversion to Flash. The high RAM socket (4F) can be converted to support Flash memory by removing CT14, CT16, CT17, and CT19 and installing CT13, CT15, and CT20. Jumpers W36 and W37 are then used for write protection. <i>Do not install W36 and W37 together. Do not install W36 or W37 when socket 4F is used for RAM (default).</i>
W37	W36	Function
IN	OUT	Write enable when socket 4F is configured as Flash
OUT	IN	Write protect when socket 4F is configured as Flash
OUT†	OUT†	Standard configuration for RAM

† Factory default.

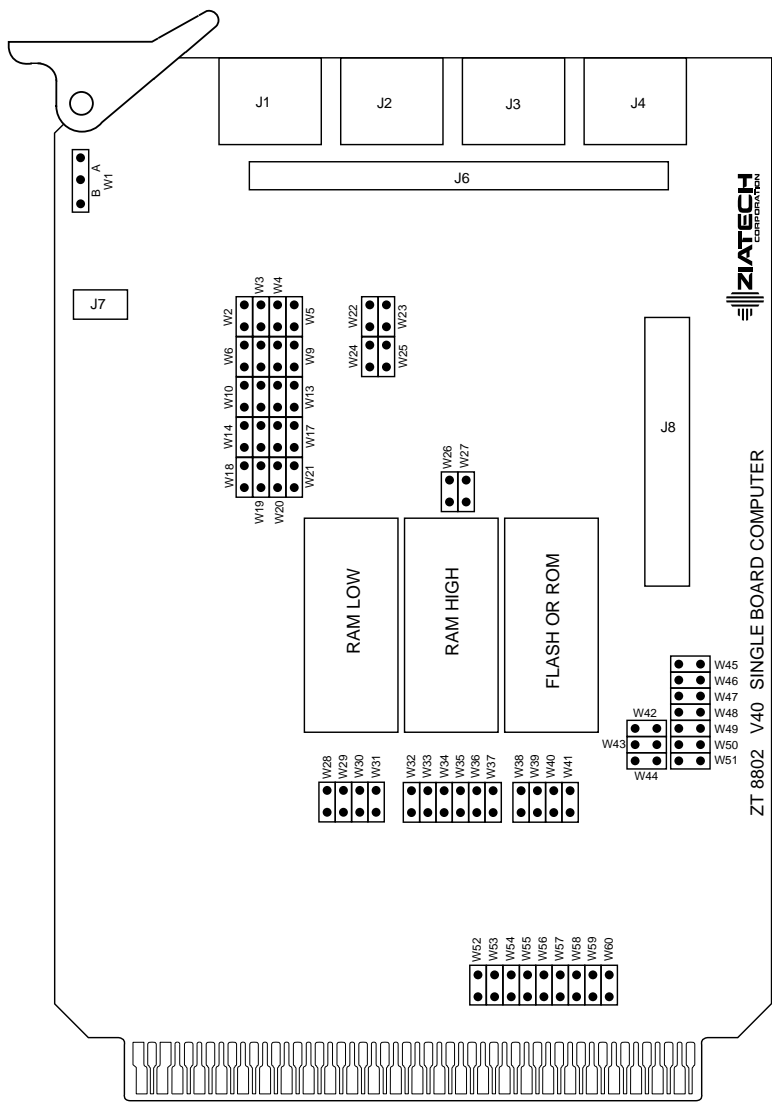
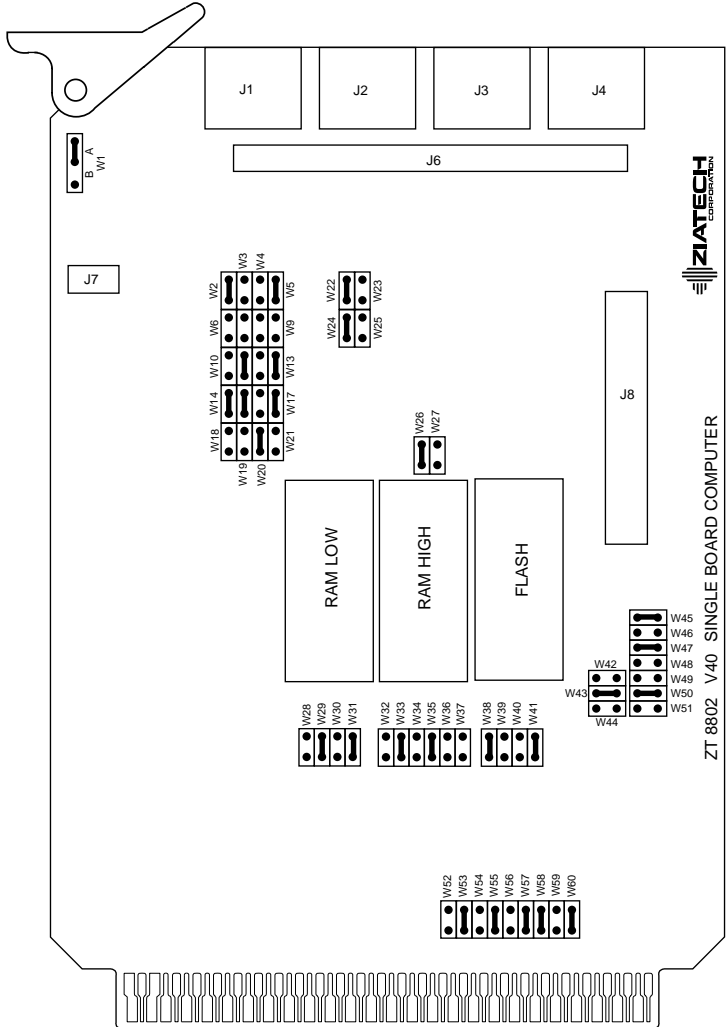


Figure A-4. Customer Jumper Configuration.



Note: INSTALL W44, W41; REMOVE W43, W40 for Flash in-circuit program capability.

Figure A-5. DOS (128K Flash/256K RAM) Default.

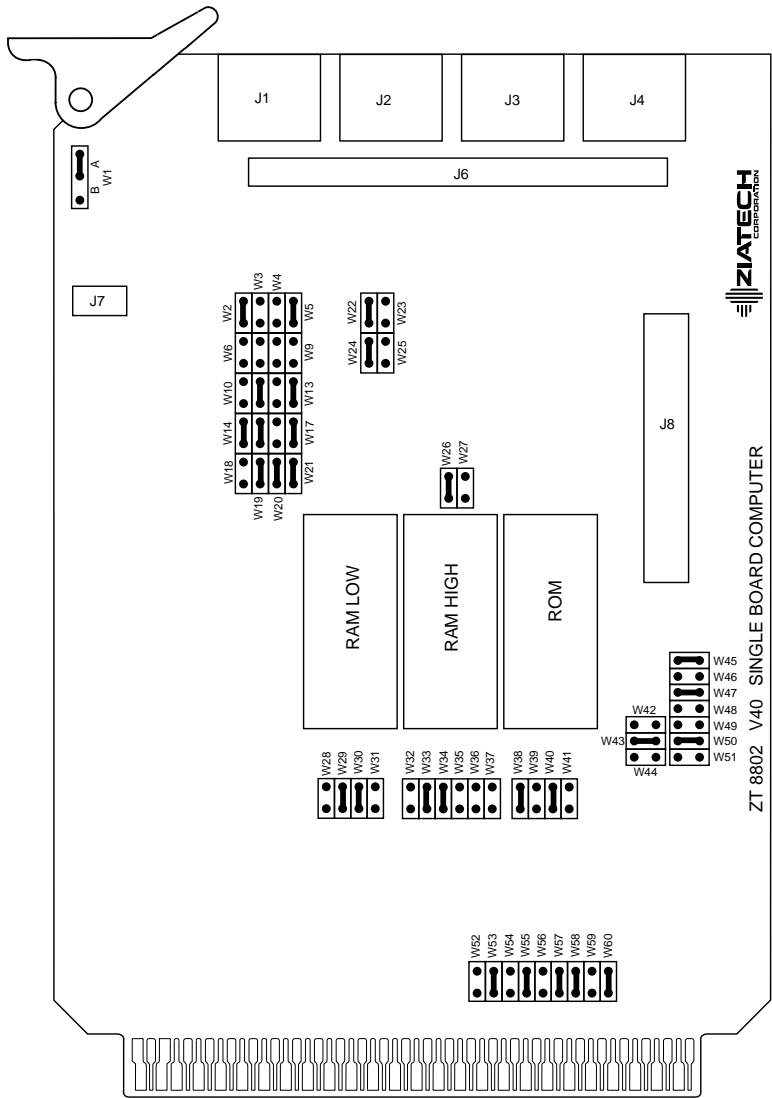


Figure A-6. DOS (512K ROM/1M RAM) Configuration.

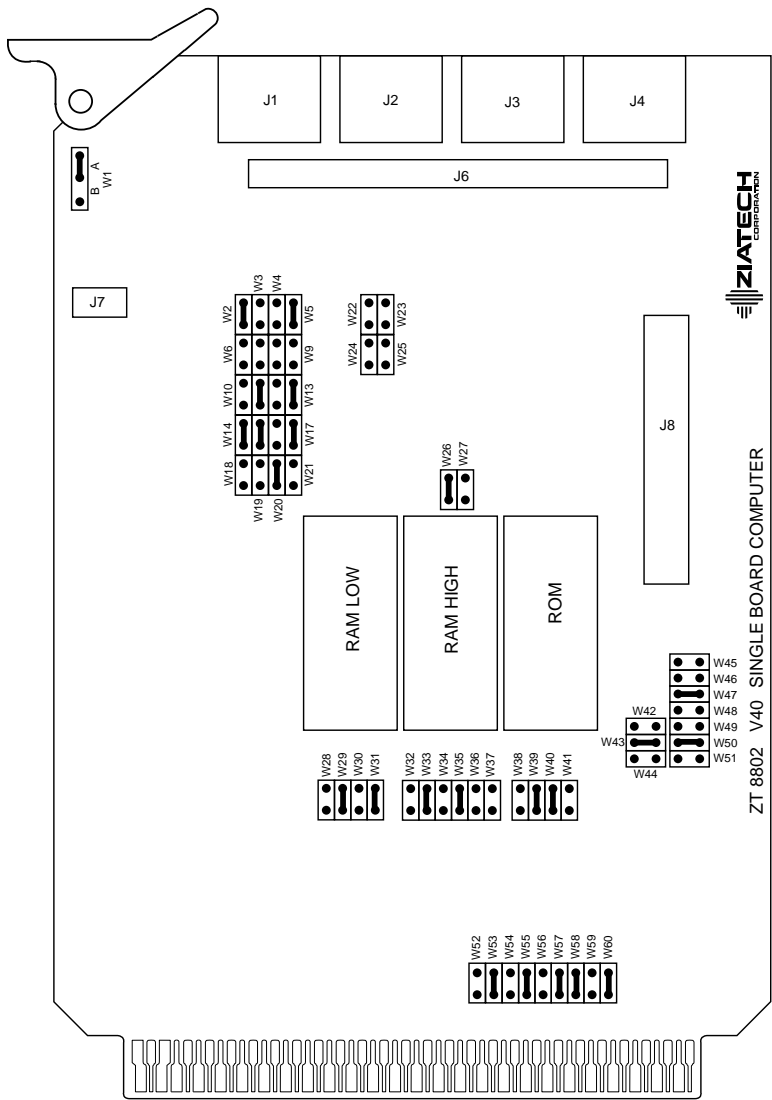


Figure A-7. STD ROM (32K ROM/128K RAM) Configuration.

Appendix B

SPECIFICATIONS

Contents	Page
ELECTRICAL AND ENVIRONMENTAL	B-2
Absolute Maximum Ratings	B-2
DC Operating Characteristics	B-2
Digital I/O Interface Characteristics	B-2
Battery Backup Characteristics	B-3
STD Bus Loading Characteristics	B-3
MECHANICAL	B-6
Card Dimensions & Weight	B-6
Connectors	B-9
Cables	B-17
Assembly Diagram	B-21

ELECTRICAL AND ENVIRONMENTAL

Absolute Maximum Ratings

Supply Voltage, Vcc	0 to 7 V
Supply Voltage, AUX +V	0 to 13 V
Supply Voltage, AUX -V	0 to -13 V

Storage Temperature

ZT 8802	-40° to +85° Celsius
ZT 88CT02 (w/o battery)	-50° to +125° Celsius
ZT 88CT02 (with battery)	-40° to +100° Celsius

Operating Temperature

ZT 8802	0° to +65° Celsius
ZT 88CT02	-40° to +85° Celsius

Non-condensing relative humidity <95% at 40°Celsius

DC Operating Characteristics

Supply Voltage, Vcc	4.75 to 5.25 V
Supply Voltage, AUX +V (Flash)	11.4 to 12.6 V
Supply Voltage, AUX -V	Not Required
Supply Current, Vcc (without SBX)	550 mA typ, 900 mA max
Supply Current, AUX +V (Flash)	0.03 A max
Supply Current, AUX -V	Not Required

Note: Current requirements measured with 256K Flash memory and 256K of RAM.

Digital I/O Interface Characteristics

VOL (max) at IOL of 12 mA (max)	0.4 V (max)
VOH (min) at IOH of 4 mA (max)	3 V (min)

Battery Backup Characteristics

Supply Voltage, Vcc	4.49 V max
Retention Time	
Real-Time Clock only	10 years min. (90,000 hrs.)
Real-Time Clock and Local RAM	12,500 hrs. min., 90,000 hrs. typical
(Assuming RAMs have combined retention current of 80 μ A)	

STD Bus Loading Characteristics

The unit load is a convenient method for specifying the input and output drive capability of STD bus cards. With this method, one unit load is equal to one LSTTL load as follows:

- Current for single input load: 20 μ A
- Current for single output drive: -400 μ A

The unit load reflects current requirements at worst case conditions over the recommended supply voltage and ambient temperature ranges. An output drive of 60 unit loads drives 60 STD bus cards having input ratings of one unit load. Table B-1 on page B-4 includes load values for the STD-80 connections. Table B-2 on page B-5 includes load values for STD 32 connections.

Specifications

Table B-1
Bus Signal Loading, P Connector.

PIN (CIRCUIT SIDE)					PIN (COMPONENT SIDE)				
OUTPUT DRIVE					OUTPUT DRIVE				
INPUT LOAD					INPUT LOAD				
MNEMONIC					MNEMONIC				
+5 VOLTS [A] GROUND DCPDN*					+5 VOLTS [A] GROUND VBATT				
	0	60	6	5	-				
D7/A23 [5]	2	58	8	7	58	2	D3/A19 [1]		
D6/A22 [5]	2	58	10	9	58	2	D2/A18 [1]		
D5/A21 [5]	2	58	12	11	58	2	D1/A17 [1]		
D4/A20 [5]	2	58	14	13	58	2	D0/A16 [1]		
A15	1	59	16	15	59	1	A7		
A14	1	59	18	17	59	1	A6		
A13	1	59	20	19	59	1	A5		
A12	1	59	22	21	59	1	A4		
A11	1	59	24	23	59	1	A3		
A10	1	59	26	25	59	1	A2		
A9	1	59	28	27	59	1	A1		
A8	1	59	30	29	59	1	A0		
RD*	1	59	32	31	59	1	WR*		
MEMRQ*	1	59	34	33	60	0	IORQ*		
(MEMEX) BHE*	0	60	36	35	60	0	IOEXP		
MCSYNC* (ALE*)	1	59	38	37	-	1	INTRQ1*		
STATUS 0*	1	59	40	39	59	1	STATUS 1*		
BUSRQ*	1	59	42	41	60	0	BUSAK* [2]		
INTRQ*	1	-	44	43	60	0	INTAK*		
NMIRQ*	1	-	46	45	59	1	WAITRQ*		
PBRESET*	1	-	48	47	60	0	SYSRESET* [3]		
INTRQ2* (CNTRL*)	1	-	50	49	60	0	CLOCK* [3]		
PCI [4]	0	-	52	51	-	0	PCO [4]		
AUX GND			54	53			AUX GND		
AUX-V			56	55			AUX+V		

Notes:

REQ indicates required connection.

[1] High order address bits multiplexed over data bus.

[2] BUSAK* is an output in permanent master configuration only.

[3] SYSRESET* and CLOCK* are outputs in permanent master configuration and inputs in temporary master configuration.

[4] PCI connected to PCO.

[5] A20 - A23 are always driven to logical 0 on the STD bus for memory cycles.

Table B-2
Bus Signal Loading, E Connector.

PIN (CIRCUIT SIDE)				PIN (COMPONENT SIDE)			
OUTPUT DRIVE				OUTPUT DRIVE			
INPUT LOAD				INPUT LOAD			
MNEMONIC				MNEMONIC			
LOCK*	0	E2	E1		GND		
XA23	0	E4	E3	0	XA19		
XA22	0	E6	E5	0	XA18		
XA21	0	E8	E7	0	XA17		
XA20	0	E10	E9	0	XA16		
RSVD	0	E12	E11	0	NOWS*		
+5V		E14	E13		+5V		
MREQX*	0	E16	E15	0	MAKx*		
GND		E18	E17		GND		
D31	0	E20	E19	0	D27		
D30	0	E22	E21	0	D26		
D29	0	E24	E23	0	D25		
D28	0	E26	E25	0	D24		
GND		E28	E27	0	D23		
D15	0	E30	E29	0	D22		
D14	0	E32	E31	0	D21		
D13	0	E34	E33	0	D20		
D12	0	E36	E35		GND		
D11	0	E38	E37	0	D19		
D10	0	E40	E39	0	D18		
D9	0	E42	E41	0	D17		
D8	0	E44	E43	0	D16		
MASTER16*	0	E46	E45		GND		
AENX*	0	E48	E47	0	IRQx		
BE3*	0	E50	E49	0	BE1*		
BE2*	0	E52	E51	0	BE0*		
GND		E54	E53	0	MEM16*		
W-R	0	E56	E55	0	M-IO		
DMAIOR*	0	E58	E57	0	DMAIOW*		
EX8*	0	E60	E59	0	IO16*		
START*	0	E62	E61	0	CMD*		
EX32*	0	E64	E63	0	EX16*		
T-C	0	E66	E65	0	EXRDY		
+5V		E68	E67	1	INTRQ3*		
DREQx*	0	E70	E69	0	DAKx*		
MSBURST*	0	E72	E71	0	SLBURST*		
XA31*	0	E74	E73	0	XA27*		
XA30*	0	E76	E75	0	XA26*		
XA29*	0	E78	E77	0	XA25*		
XA28*	0	E80	E79	0	XA24*		

Note: REQ indicates required connection.

MECHANICAL

Card Dimensions & Weight

The ZT 8802 meets the *STD-80 Series Bus Specification and Designer's Guide* for all mechanical parameters. In a card cage with 0.625 inch spacing, the ZT 8802 requires one card slot without an SBX expansion module installed and two card slots with an SBX expansion module installed. Mechanical dimensions are shown in Figure B-1 and outlined below.

Board Length	16.51 ±0.063 cm (6.500 ±0.025 in)
Board Width	11.43 ±0.038 cm (4.500 ±0.015 in)
Board Thickness	0.158 ±0.013 cm (0.062 ±0.007 in)
Board Weight	200 g (7 oz)
Board Height From Top Surface	9.652 mm (0.38 in) max.
Including Battery	12.192 mm (0.48 in) max.
Including SBX Expansion Module .	2.794 cm (1.10 in) max.
Board Height From Bottom Surface	1.143 mm (0.045 in) max.

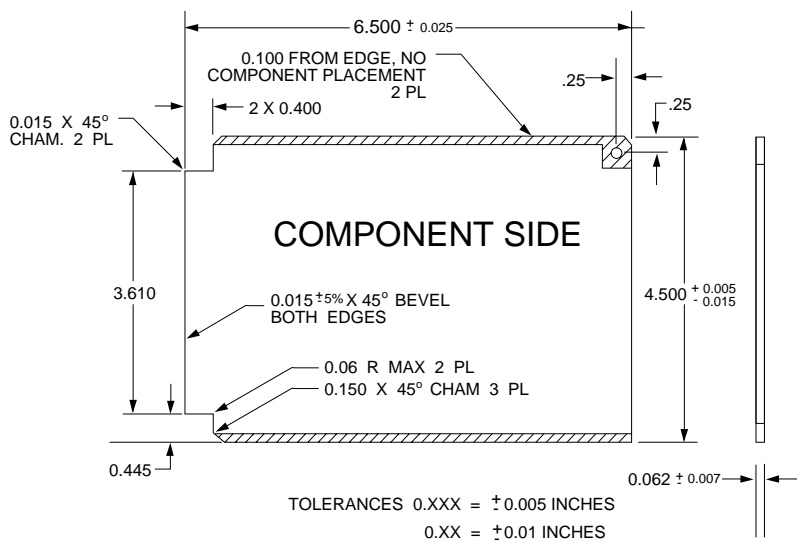


Figure B-1. ZT 8802 Card Dimensions.

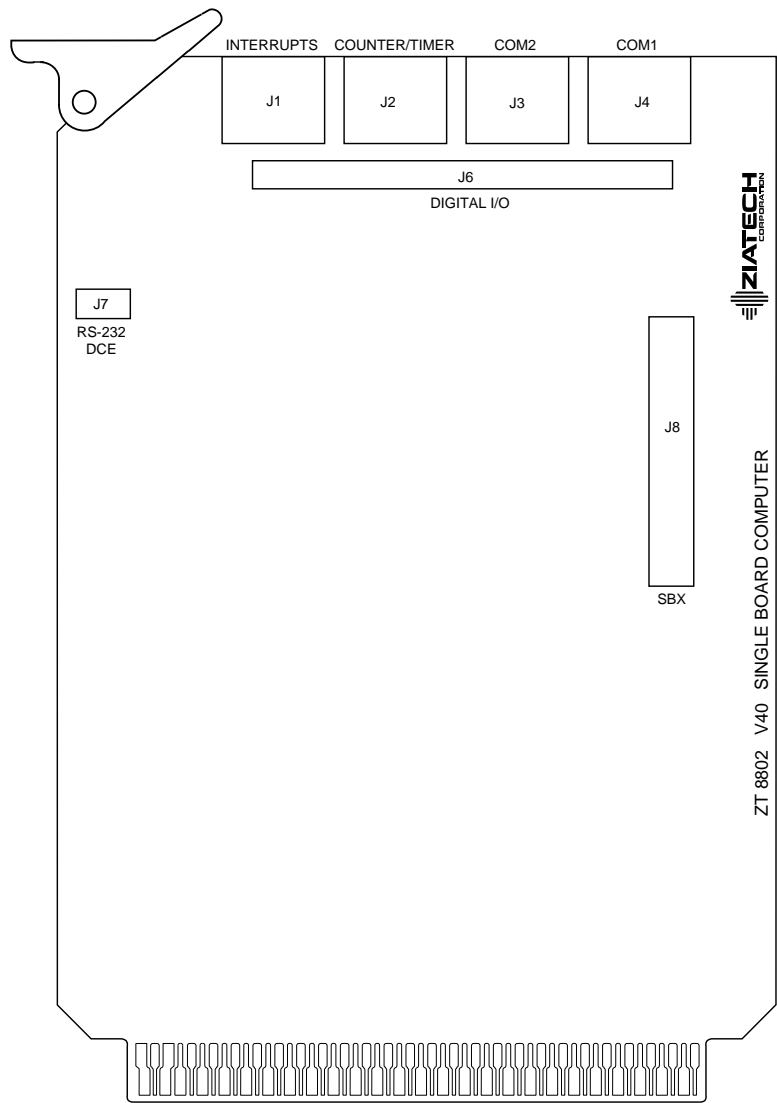


Figure B-2. Connector Locations.

Connectors

The ZT 8802 includes 10 connectors to interface to the STD bus and application-specific devices (see Figure B-2). The following pages describe the connectors and list the pin assignments for each.

- P:** The P connector is the interface between the ZT 8802 and the STD-80 bus. This connector is a 56-pin (dual 28-pin) card-edge connector with fingers on 0.125 inch centers. The mating connector is a Viking 3VH28/1CNK5 or equivalent for the solder tail, or a Viking 3VH28/1CND5 or equivalent for a three-level wire wrap. The pin assignments for the P connector are shown on page B-4.
- E:** The E connector extends the P connector to interface the ZT 8802 to the STD 32 bus. This connector combines with the P connector to make a 114-pin (dual 57-pin) card-edge connector with fingers on 0.062 inch centers. The mating connector is a Viking S3VT68/5DP12 or equivalent for the solder tail, or a Viking S3VT68/5DE12 or equivalent for the card extender. The pin assignments for the E connector are shown on page B-5.

Figure B-3 on page B-10 shows the P and E connector pinouts.

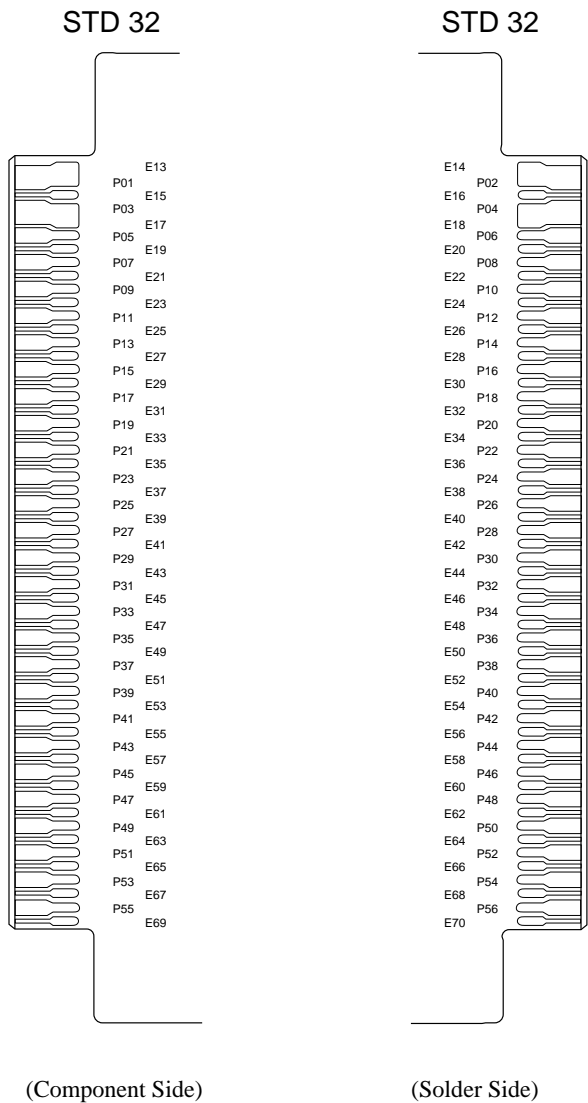


Figure B-3. P/E Connector Pinout.

J1: J1 is a latching 10-pin (dual 5-pin) male transition connector with 0.1 inch lead spacing. The interrupt inputs are available through this connector. Table B-3 lists the pin assignments. The mating connector is a T&B Ansley #622-1000 or equivalent.

Table B-3
J1 Pin Assignments.

Pin	Description	Signal
2	Frontplane Interrupt	$\overline{\text{FP1}}$
4	"	$\overline{\text{FP3}}$
6	"	$\overline{\text{FP5}}$
8	"	$\overline{\text{FP6}}$
10	"	$\overline{\text{FP7}}$
Odd	Ground	

Specifications

J2: J2 is a latching 10-pin (dual 5-pin) male transition connector with 0.1 inch lead spacing. The counter/timer inputs are available through this connector. Table B-4 lists the pin assignments. The mating connector is a T&B Ansley #622-1000 or equivalent.

Table B-4
J2 Pin Assignments.

Pin	Description
2	Timer/Counter Clock
4	Counter 2 $\overline{\text{CONTROL}}$
6	Counter 2 Output
8	No Connection
10	No Connection
Odd	Ground

J3, J4: J3 and J4 are latching 10-pin (dual 5-pin) male transition connectors with 0.1 inch contact spacing. These connectors include the RS-232-C serial interface signals for COM2 and COM1, respectively. Table B-5 lists the pin assignments. The mating connector is a T&B Ansley #622-1030 or equivalent.

Table B-5
J3 and J4 Pin Assignments.

Pin#	Signal	Type	Description
1	DCD	In	Data Carrier Detect
2	DSR	In	Data Set Ready
3	RXD	In	Receive Data
4	RTS	Out	Request To Send
5	TXD	Out	Transmit Data
6	CTS	In	Clear To Send
7	DTR	Out	Data Terminal Ready
8	RI	In	Ring Indicator
9	GND	---	Signal Ground
10	VCC	---	+5 V

J5: J5 is reserved for factory use.

J6: J6 is a 56-pin (dual 28-pin) vertical male header with 0.1 inch lead spacing. This connector provides 48 digital I/O lines, fused +5 V $\pm 10\%$, and ground. Table B-6 on page B-14 lists the pin assignments. J6 can interface directly to the 56-pin header on the ZT 2226 I/O module mounting rack via a standard 56-conductor ribbon cable such as the ZT 90089, shown on page B-18. J6's pin assignments enable the ZT 2225 cable to connect J6 directly to one or two I/O module mounting racks with 8, 16, or 24 positions. For applications not using this cable, the mating connector is a T&B Ansley #622-5600 or equivalent.

Table B-6
J6 Parallel Port Pinout.

Pin	Signal	Port Address [hex]	Pin	Signal	Port Address [hex]
1	IO00/E0	FA00 bit 0	28	IO24	FA03 bit 0
2	IO01/E1	FA00 bit 1	29	IO25	FA03 bit 1
3	IO02/E2	FA00 bit 2	30	IO26	FA03 bit 2
4	IO03/E3	FA00 bit 3	31	IO27	FA03 bit 3
5	IO04/E4	FA00 bit 4	32	IO28	FA03 bit 4
6	IO05/E5	FA00 bit 5	33	IO29	FA03 bit 5
7	IO06/E6	FA00 bit 6	34	IO30	FA03 bit 6
8	IO07/E7	FA00 bit 7	35	IO31	FA03 bit 7
9	GND	--	36	GND	--
10	IO08	FA01 bit 0	37	IO32	FA04 bit 0
11	IO09	FA01 bit 1	38	IO33	FA04 bit 1
12	IO10	FA01 bit 2	39	IO34	FA04 bit 2
13	IO11	FA01 bit 3	40	IO35	FA04 bit 3
14	IO12	FA01 bit 4	41	IO36	FA04 bit 4
15	IO13	FA01 bit 5	42	IO37	FA04 bit 5
16	IO14	FA01 bit 6	43	IO38	FA04 bit 6
17	IO15	FA01 bit 7	44	IO39	FA04 bit 7
18	GND	--	45	GND	--
19	IO16	FA02 bit 0	46	IO40	FA05 bit 0 (DC/DC)
20	IO17	FA02 bit 1	47	IO41	FA05 bit 1 (OSC)
21	IO18	FA02 bit 2	48	IO42	FA05 bit 2 (WDSTB)
22	IO19	FA02 bit 3	49	IO43	FA05 bit 3 (Reserved)
23	IO20	FA02 bit 4	50	IO44	FA05 bit 4 (RTC)
24	IO21	FA02 bit 5	51	IO45	FA05 bit 5 (MD0)
25	IO22	FA02 bit 6	52	IO46	FA05 bit 6 (MD1)
26	IO23	FA02 bit 7	53	IO47	FA05 bit 7 (LED)
27	GND	--	54	+5 V	-- 1 A fused
			55	+5 V	-- 1 A fused
			56	+5 V	-- 1 A fused

J7: J7 is a latching 3-pin low profile header with 0.1 inch lead spacing. It provides RS-232-C DCE communications (when used with the ZT 90069 cable). Table B-7 lists the pin assignments.

Table B-7
J7 Pin Assignments.

Pin	Description
1	RS-232-C Receive Data (RxD)
2	RS-232-C Transmit Data (TxD)
3	Ground

J8: J8 is a latching 36-pin (dual 18-pin) female SBX connector with 0.1 inch lead spacing. This connector includes the power, address, data, and control signals needed to add custom I/O to the ZT 8802. Table B-8 on page B-16 lists the pin assignments. The mating connector for an 8-bit SBX interface is a T&B Ansley #609 BX360.

J9: J9 is reserved for factory use.

Table B-8
J8 SBX Expansion Module Pinout.

Pin	Signal ^[1]	Description	Pin	Signal ^[1]	Description
1	+12V	+12 V	19	MD7	Data Bit 7
2	-12V	-12 V	20	MCS1*	Chip Select 1 ^[2]
3	GND	Signal Ground	21	MD6	Data Bit 6
4	+5V	+5 V	22	MCS0*	Chip Select 0 ^[2]
5	RESET	Reset	23	MD5	Data Bit 5
6	MCLK	10 MHz Clock	24	RSVD	Reserved - Address 6 ^[4]
7	MA2	Address 2	25	MD4	Data Bit 4
8	MPST*	Module present ^[3]	26	TDMA	Terminate DMA ^[3]
9	MA1	Address 1	27	MD3	Data Bit 3
10	RSVD	Reserved - Address 5 ^[4]	28	OPT1	Option 1 - Address 4 ^[4]
11	MA0	Address 0	29	MD2	Data Bit 2
12	MINTR1	Interrupt 1 ^[5]	30	OPT0	Option 0- Address 3 ^[4]
13	IOWRT*	I/O Write	31	MD1	Data Bit 1
14	MINTR0	Interrupt 0 ^[5]	32	MDACK*	DMA Acknowledge ^[3]
15	IORD*	I/O Read	33	MD0	Data Bit 0
16	MWAIT*	Wait Request	34	MDRQT	DMA Request ^[3]
17	GND	Ground	35	GND	Ground
18	+5V	+5 V	36	+5V	+5 V

Notes:

- [1] Signals ending with an asterisk (*) are active low, and signals without an asterisk are active high.
- [2] The 8-bit I/O address range for chip select 0 is FB00h through FB7Fh; for chip select 1, FB80h through FBFFh.
- [3] These signals are not supported. The MPST* is not connected and TDMA is connected to ground. MDACK* is passively pulled up.
- [4] These signals provide additional address lines to the three supported in the expansion module specification. This feature is supported with CT9 through CT12.
- [5] Interrupt 0 and interrupt 1 are routed to the interrupt jumper block.

Cables

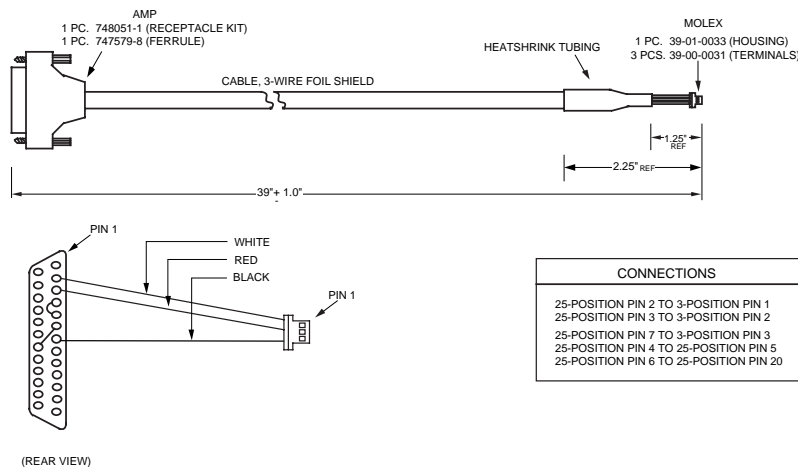


Figure B-4. ZT 90069 Revision B Serial Cable.

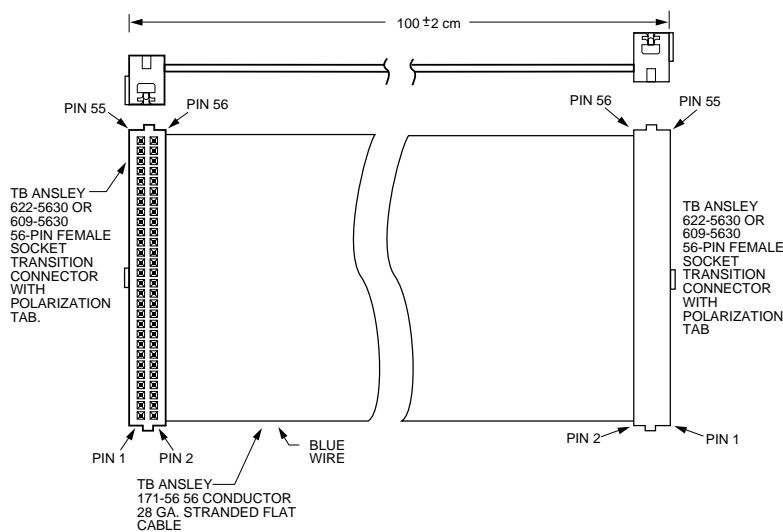


Figure B-5. ZT 90089 Revision A Digital I/O Cable.

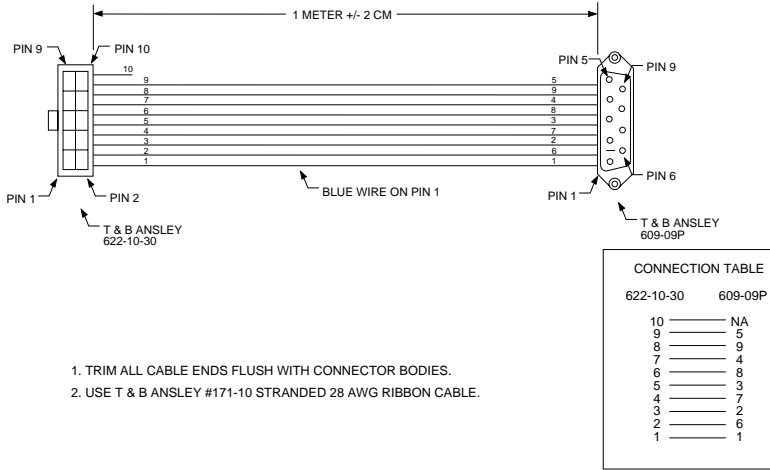


Figure B-6. ZT 90136 Revision 0 Serial Cable.

Specifications

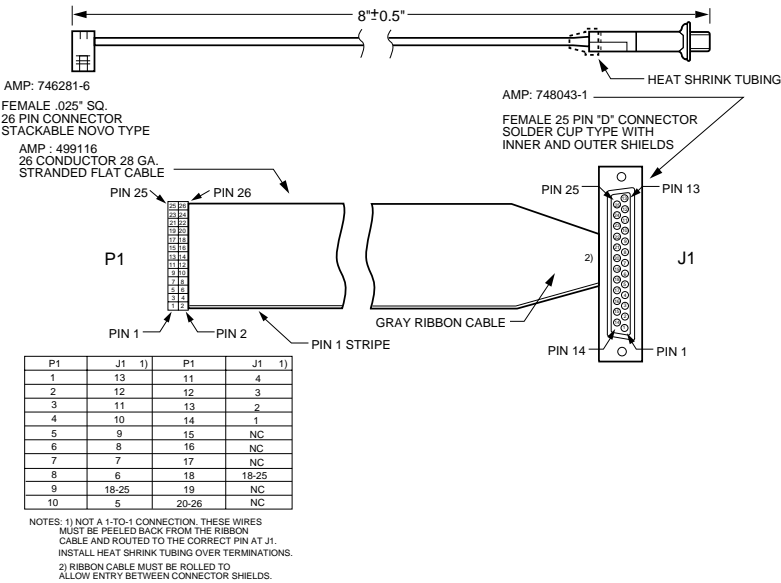


Figure B-7. ZT 90156 Revision 0 Printer Interface Cable.

PIA SYSTEM SETUP CONSIDERATIONS

Contents	Page
OVERVIEW	C-1
PREVENTING SYSTEM LATCHUP	C-2
POWER SUPPLY SEQUENCE MISMATCH	C-2
SIGNAL LEVEL MISMATCH	C-5

OVERVIEW

The 16C49 Parallel Interface Adapter (PIA) device used on the ZT 8802 is designed by Ziatech to offer bidirectional I/O signals with or without event sense capability. This device features low-power, high-speed, wide temperature operation achievable only by utilizing CMOS technology.

Although CMOS technology offers many advantages, you must observe a few cautions when interfacing to any CMOS parts.

CMOS inputs and outputs can exhibit latchup characteristics. These inherent characteristics of any CMOS technology can result in the formation of a Silicon-Controlled Rectifier (SCR) that appears between Vcc and ground when voltages greater than Vcc or less than ground are applied to inputs or outputs. When this happens, Vcc is effectively shorted to ground. The only way to remove the latchup condition is to shut off the power supply. If a large current is allowed to flow through the chip, its operating temperature may increase, it may exhibit intermittent operation, or it may be damaged.

The purpose of this appendix is to illustrate precautions you should take to prevent latchup conditions.

PREVENTING SYSTEM LATCHUP

The most common causes of latchup are:

- Input signals applied before the input circuitry is powered, resulting in a signal to power supply sequence mismatch
- Input signals greater than V_{cc} or less than ground, resulting in a signal level mismatch

Each of these conditions is covered in the following pages.

POWER SUPPLY SEQUENCE MISMATCH

A common application is to interface to a 24-position ZT 2226, Opto 22, or equivalent I/O module rack. V_{cc} and ground are provided from the ZT 8802 through connector J6, with V_{cc} protected by a 1 A fuse. (See page B-14 for J6 pinouts.) This application is illustrated in Figure C-1. In this application, no power supply sequence mismatch exists because the power supplying the input circuitry within the PIA is applied before or at the same time as the power supplying the external signals. Proper system operation will result.

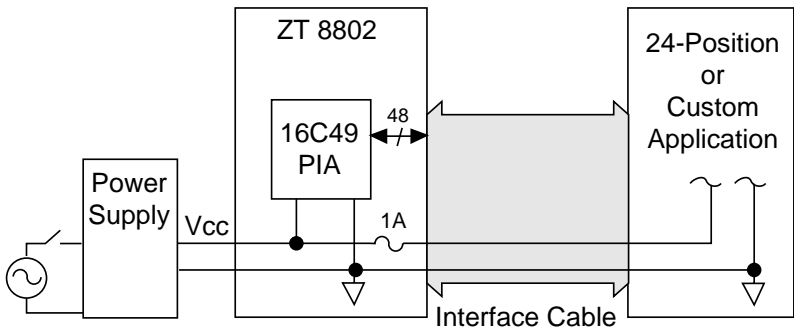
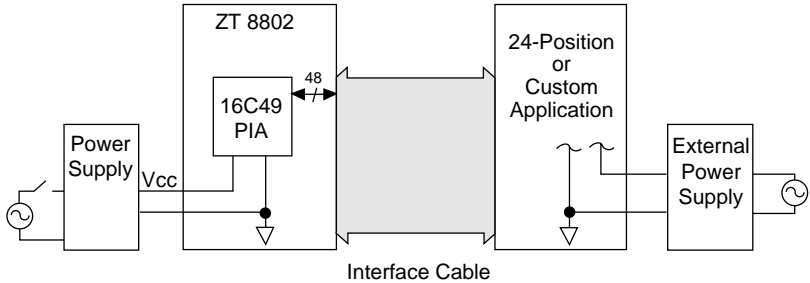


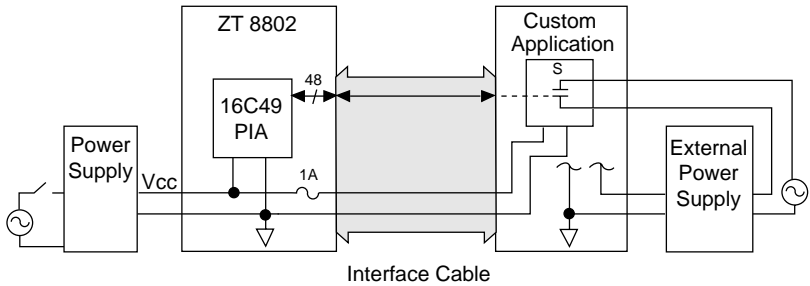
Figure C-1. I/O Rack V_{cc} and Ground Supplied Via Interface Cable. Correct Power Supply Sequence, Signal Level Matched.

However, if a power source other than that supplying the PIA is used to power the external signals, then a power sequence mismatch could occur, resulting in a latchup condition. An external power source might be required if the external circuitry requires more than the 1 A supplied by the cable or if a custom interface is being designed (see Figure C-2).



*Figure C-2. I/O Rack Vcc and Ground Supplied Externally.
Potential Power Supply Sequence Mismatch, Signal Level Mismatch.*

One solution is to switch the external signals' power supply with an output that is controlled by the computer. In this manner, if the computer is off, so is the external power supply. This solution is illustrated in Figure C-3.



*Figure C-3. Computer-Switched External Power Supply.
Correct Power Supply Sequence, Potential Signal Level Mismatch.*

PIA System Setup Considerations

A simpler solution is to power the relay controlling the external power supply directly from Vcc and ground supplied by the interface cable.

Another solution is to use the same switch to control the computer's power supply and the external signals' power supply (see Figure C-4). This is an acceptable solution for power supply sequence mismatches as long as the computer supply ramps up faster than the external power supply. This ensures that the PIA input circuitry is powered before the external signal circuitry.

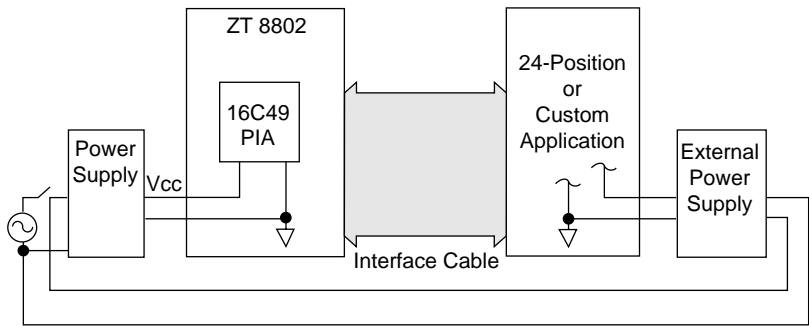


Figure C-4. Computer & Ext. Power Supply w/ Common Switch. Correct Power Supply Sequence, Potential Signal Level Mismatch.

SIGNAL LEVEL MISMATCH

Power supplying the external signal in Figure C-1 is always relative to the PIA input circuitry power because power is provided over the interface cable. Signal level mismatches will not occur and proper system operation will result. However, if separate power supplies are used, there are two predominant causes of signal level mismatches.

The first (assuming no sequencing problems) occurs when the two supplies are not referenced to each other, as illustrated in Figures C-2 through C-4. This results in signals that may be higher than V_{cc} or lower than ground, potentially causing SCR latchup. All that is generally needed is to reference one supply to the other, typically by connecting a common ground. The most convenient way of connecting a common ground is to use the interface cable. Figures C-5 through C-7 illustrate correct ground connections.

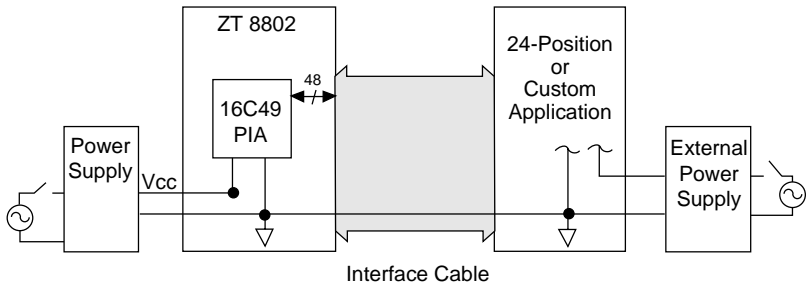
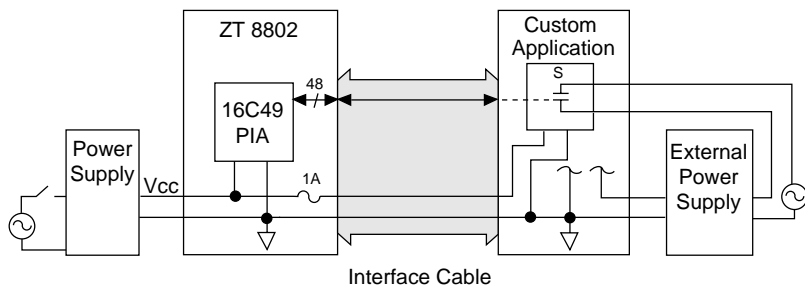
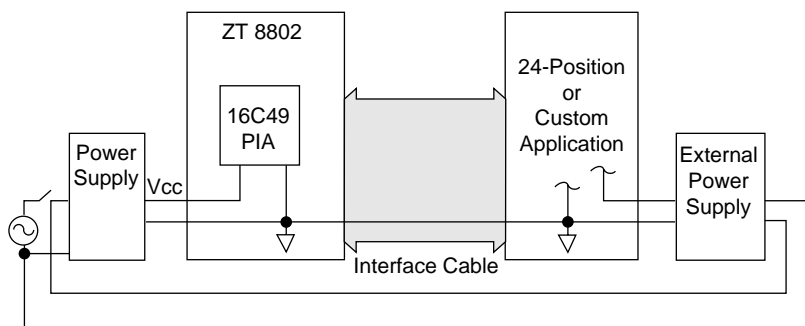


Figure C-5. I/O Rack V_{cc} Supplied Externally, Common Ground. Potential Power Supply Seq. Mismatch, Correct Signal Level Match.

PIA System Setup Considerations



*Figure C-6. Computer-Switched External PS, Common Ground.
Correct Power Supply Sequence, Correct Signal Level Match.*



*Figure C-7. Computer & External PS w/ Common Switch & Ground.
Correct Power Supply Sequence, Correct Signal Level Match.*

The second cause of mismatch occurs when the two power supplies are referenced to each other but the V_{cc} difference between the two power supplies exceeds .5 V. This results in signals that could be greater than V_{cc} , causing SCR latchup. This is easily remedied by adjusting the external power supply voltage to be within .5 V of the computer power supply voltage.

You can find additional design information in the *Advanced CMOS Logic Designer's Handbook*, published by Texas Instruments.

Appendix D

CUSTOMER SUPPORT

Contents	Page
OVERVIEW	D-1
REVISION HISTORY	D-2
TECHNICAL ASSISTANCE	D-2
RELIABILITY	D-3
RETURNING FOR SERVICE	D-4
ZIATECH WARRANTY	D-5

OVERVIEW

This appendix offers a product revision history, technical assistance for the ZT 8802, and the necessary information should you need to return your ZT 8802 for repair.

REVISION HISTORY

Revision 0 is the first production release of the ZT 8802.

TECHNICAL ASSISTANCE

If you have a technical question, please call Ziatech's Customer Support Service at one of the following numbers.

Corporate Headquarters: (805) 541-0488
(805) 541-5088 (FAX)

RELIABILITY

Ziatech has taken extra care in the design of the ZT 8802 to ensure reliability. The four major ways in which reliability is achieved are:

1. The product is designed with the latest hardware and software techniques. All device interconnections undergo extensive timing and parametric analysis over the entire specified operating range.
2. The advanced low-power Schottky TTL devices and high speed CMOS TTL devices used in the ZT 8802 are high-reliability parts available from several manufacturers.
3. Ziatech burns in each board under power and tests it to ensure that the infant mortality phase is passed before the product is shipped.
4. Each ZT 8802 is marked with an identification number used to track the product. Any negative reliability trends are identified and corrected immediately.

RETURNING FOR SERVICE

Before returning any of Ziatech's products, you must phone Ziatech at (805) 541-0488 and obtain a Returned Material Authorization (RMA) number. The following information is needed to expedite the shipment of a replacement to you:

1. Your company name and address for invoice
2. Shipping address and phone number
3. Product I.D. number
4. If possible, the name of a technically qualified individual at your company familiar with the mode of failure on the board

If the unit is out of warranty, service is available at a predesignated service charge. Contact Ziatech for pricing and please supply a purchase order number for invoicing the repair.

Pack the ZT 8802 in *anti-static* material and ship in a sturdy cardboard box with enough packing material to adequately cushion the board. *Any product returned to Ziatech improperly packed will immediately void the warranty for that particular product!* Mark the RMA number clearly on the outside of the box before returning.

ZIATECH WARRANTY

FIVE-YEAR LIMITED WARRANTY

Products manufactured by Ziatech Corporation are covered from the date of purchase by a five-year warranty against defects in materials, workmanship, and published specifications applicable to the date of manufacture. During the warranty period, Ziatech will repair or replace, solely at its option, defective units provided they are returned at customer expense to an authorized Ziatech repair facility. Products which have been subjected to misuse, abuse, neglect, alteration, or unauthorized repair, determined at the sole discretion of Ziatech, whether by accident or otherwise, are excluded from warranty. The warranty on fans and disk drives is limited to two years, the warranty on software is limited to one year, and the warranty on flat panel displays is limited to nine months from date of purchase. Other products and accessories not manufactured by Ziatech are limited to the warranty provided by the original manufacturer. Consumable items (fuses, disk media, batteries, etc.) are not covered by this warranty.

Ziatech may offer, where applicable and available, replacement products; otherwise, repairs requiring components, assemblies, and other purchased materials may be limited by market availability.

Ziatech assumes no liability resulting from changes to government regulations affecting use of materials, equipment, safety, and methods of repair. Ziatech may, at its discretion, offer replacement products.

The above warranty is in lieu of any other warranty, whether expressed, implied, or statutory, including, but not limited to, any warranty for fitness of purpose, merchantability, or freedom from infringement or the like, and any warranty otherwise arising out of any proposal, specifications, or sample. Ziatech neither assumes nor authorizes any person to assume for it any other liability. The liability of Ziatech under this warranty agreement is limited to a refund of the purchase price. In no event shall Ziatech be liable for loss of profits, use, incidental, consequential, or other damage, under this agreement.

LIFE SUPPORT POLICY

Ziatech products are not authorized for use as critical components in life support devices or systems without the express written approval of the president of Ziatech Corporation. As used herein:

1. Life support devices or systems are devices or systems which support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be expected to cause the failure of the life support device or system, affect its safety, or limit its effectiveness.

GLOSSARY

Contents	Page
OVERVIEW	E-1
TERMINOLOGY	E-1

OVERVIEW

This appendix defines important terms and acronyms used throughout this manual.

TERMINOLOGY

ASIC Application Specific Integrated Circuit. The 16C49 device used in the ZT 8802 is a custom ASIC device designed by Ziatech to implement 48 points of parallel I/O.

backplane The edge of the board that inserts into the STD bus connector. This term is generally used to define the location of signals that are routed across the STD bus.

BAU Bus Arbitration Unit. Section of the CPU that controls which internal or external bus master has access to the buses at any given time.

Glossary

BCD	Binary Coded Decimal. Representation of the cardinal numbers 0 through 9 by ten binary codes. Each binary code is 4 binary digits long.
BIU	Bus Interface Unit. Section of the CPU that controls the external address, data, and control buses.
BUSAK*	Bus Acknowledge. STD bus signal (pin 41) that indicates the bus is available for use by a requesting controller.
BUSRQ*	Bus Request. STD bus signal (pin 42) that causes the controlling processor to suspend operations on the STD bus by releasing all three-state STD bus lines for use by another processor.
CGU	Clock Generator Unit. Section of the CPU that provides a clock reference with a 50% duty cycle to the CPU.
CMOS	Complementary Metal Oxide Semiconductor. Provides low power density and high noise immunity.
CNTRL*	Control. This STD bus signal (pin 50) was used in previous designs for special clock timing on peripheral boards. It may also be used as an interrupt request, INTRQ2*, on the backplane.

DCE	Data Communication Equipment. One of two possible orientations (DCE or DTE) for drivers and receivers in the RS-232-C serial communications protocol.
DCPDN*	DC Power-Down. STD bus signal (pin 6) that indicates DC power is dropping.
DCU	DMA Control Unit. Section of the CPU that controls high speed data transfer between I/O and memory devices.
DMA	Direct Memory Access. Used for faster data transfer rate when processing of I/O data on a byte-by-byte basis is not required.
DTE	Data Terminal Equipment. One of two possible orientations (DCE or DTE) for drivers and receivers in the RS-232-C serial communications protocol.
EOI	End Of Interrupt. A byte sent to the interrupt controller that signifies the interrupt level has been serviced.
EPROM	Erasable Programmable Read-Only Memory. Memory is programmed and erased with ultra-violet light.
Flash	A memory device that is written to under program control, similar to a RAM device, and that retains data even if power is removed, similar to a ROM device.

frontplane	The edge of the board on which the extractor is located, opposite to the backplane. This term is generally used to define the location of user interface signals.
ICU	Interrupt Control Unit, on the V40 CPU.
INTRQ* INTRQ1* INTRQ2* INTRQ3*	Interrupt Requests. These STD-80 signals are processor card input signals that conditionally interrupt the program when enabled by a specific program instruction. INTRQ2* was formerly called CNTRL*; see <i>CNTRL*</i> .
IOEXP	Input/Output Expansion. STD-80 control signal used to expand the I/O address capability or to determine the validity of an I/O access to the board.
LSTTL	Low Power Schottky Transistor Transistor Logic. See <i>Schottky TTL</i> .
mark	A negative voltage on a serial link.
MEMEX	Memory Expansion. STD bus signal (pin 36) used to expand the memory address capability.
NMI	Non-Maskable Interrupt. Interrupt request input that cannot be disabled through software control. Generally used to signal events such as power failure and parity error.
NMIRQ*	Non-Maskable Interrupt Request. STD bus signal (pin 46) that generates an NMI. See <i>NMI</i> .

PCA	PC-Assisted. Software development system that provides an IBM PC-based development tool for application programming and debugging.
PIA	Parallel Interface Adapter. The 16C49 ASIC used on the ZT 8802 is a 48-line parallel interface adapter and is sometimes referred to as a PIA.
PIC	Programmable Interrupt Controller (on the ZT 8802, equivalent to the Intel 8259A). This device prioritizes and handles interrupt requests from the STD bus.
pop	A stack operation that retrieves one byte from the top of the processor stack.
prefetch	Instructions are fetched and stored into a queue on the microprocessor prior to execution in order to optimize performance.
push	A stack operation that stores one byte onto the top of the processor stack.
RD*	Read from memory or I/O. STD-80 bus signal (pin 32) that indicates the processor or another bus-controlling device needs to read data from memory or from an I/O device.
RS-232-C	An acronym for Required Standard 232 of the Electronics Industry Association. Interface standard between Data Terminal Equipment and Data Communication Equipment, employing serial binary data exchange.

Schottky TTL	Provides high speed, low noise, and compatibility with standard TTL. See <i>TTL</i> .
SCR	Silicon-Controlled Rectifier. Equivalent to a reverse-blocking semiconductor. In CMOS devices, an SCR can form between Vcc and ground under certain conditions. See <i>CMOS</i> and Appendix C.
SCU	Serial Control Unit. Single asynchronous serial channel used for serial communication between the CPU and a serial device external to the CPU.
space	A positive voltage on a serial link.
TCU	Timer/Counter Control Unit on the V40 CPU.
TTL	Transistor-Transistor Logic.
UART	Universal Asynchronous Receiver/Transmitter. Used for serial communications to external devices.
VCR	V40 Configuration Registers. Twelve programmable registers used to configure the CPU to meet the needs of varying applications.
WCU	Wait Control Unit. Section of the CPU that can define a different number of wait states for each of the three areas of the memory space.

WR*	Write to memory or I/O. Output to the STD-80 bus (pin 31). Clocks data to memory or output port latches.
XTP	Stands for extended temperature operation in ambient temperatures of -40° to +85° Celsius.

INDEX

- A -

absolute maximum ratings	B-2
ASIC (16C49)	13-3
definition	E-1
assembly diagram	B-21
automation engine	2-18

- B -

backplane, definition	E-1
battery backup	1-3, 1-8, 2-13, 3-35
characteristics	B-3
jumper configuration	A-3
battery life	3-35
BAU - Bus Arbitration Unit	4-23
definition	E-1
baud rate	5-5, 5-13
16C452 serial ports	10-22, 10-23
V40 serial port	9-9
Binary Coded Decimal (BCD), definition	E-2
BIU - Bus Interface Unit	4-22
definition	E-2
block diagram	
16C452 serial port	10-8
ZT 8802	1-6
board dimensions	B-7
break interrupt (BI) indicator	10-26

Index

bus loading characteristics	B-3
BUSRQ*/BUSAK*	1-8, 3-14, 8-3
definitions	E-2
handshake	4-3

- C -

cables	
ZT 90022 digital I/O	13-14
ZT 90072 digital I/O	13-14
ZT 90089 digital I/O	13-14, B-18
ZT 90137 digital I/O	13-14
ZT 90156 printer interface	13-18, B-20
ZT 90069 serial I/O	2-10, 2-13, 3-23, B-17
ZT 90136 serial I/O	2-13, B-19
cascaded interrupts	3-29
cascade mode	8-10
Centronics printer interface	13-18
CGU - Clock Generator Unit	4-23
definition	E-2
clear to send (CTS*)	10-9, 10-32
clock/calendar (see real-time clock)	14-1
clock shutdown	1-11, 3-38
CMOS version	1-2, 2-3, 2-5, 3-37
added features	3-37
cautions when using CMOS parts	C-1
definition of CMOS	E-2
halt with interrupt restart	3-38
CNTRL* (INTRQ2*)	1-11
definition	E-2
command byte (real-time clock)	14-5
commonly asked questions	3-4
communications element (VL 16C452)	10-2
block diagram	10-8
complementary metal oxide semiconductor (see CMOS)	C-1
connectors	
J2 (counter/timer)	6-4, A-13, B-12

J6 (digital I/O)	3-22, 11-2, 13-4, 13-14, B-13
J1 (interrupt inputs)	B-11
J5 (reserved)	B-13
J9 (reserved)	B-15
J7 (RS-232-C DCE)	2-13, 9-3, B-15
J3 (RS-232-C interface, COM2)	2-13, B-13
J4 (RS-232-C interface, COM1)	2-13, B-13
J8 (SBX expansion module)	12-4, B-15
locations	B-8
P/E (backplane interface)	B-9
pin assignments	B-9
control port	3-31
bit map	3-32, 13-6
configuration	13-5, A-10
cooling systems	2-5
counter/timers	1-10, 6-1
clock selection register (TCKS)	5-12
connector (J2)	6-4, A-13, B-12
effect of keyboard error on Counter/Timer 2 frequency	6-4
programmable registers	6-5
CPU - Central Processing Unit	4-6
block diagram	4-7
registers	4-6
reset	4-26
standby mode	4-21
customer support	D-1
cuttable traces	
CT1, 2 (factory options)	A-14
CT13-17, 19-20 (factory options)	A-15

- D -

daisy-chaining I/O module mounting racks	13-14
data communication equipment (DCE)	1-9, 3-23
definition	E-3
data ready (DR) indicator	10-25

Index

data set ready (DSR*)	10-9, 10-33
data terminal equipment (DTE)	1-9, 3-23
definition	E-3
data terminal ready (DTR*)	10-10, 10-30
DBA/DCA - DMA Base and Current Address registers	8-8
DBC/DCC - DMA Base and Current Count registers	8-7
DCH - DMA Channel register	8-5
DC operating characteristics	B-2
DCPDN*	1-8, 3-31
definition	E-3
DC power fail	1-9, 3-34
DCU - DMA Control Unit	4-25
cascade mode	8-10
definition	E-3
I/O port addressing	8-4
programmable registers	8-4
DDC - DMA Device Control register	8-9
debugging monitor	2-9
delta clear to send (DCTS) indicator	10-32
delta data set ready (DDSR)	10-32
delta received line signal detector (DRLSD)	10-32
development software	2-6
device driver package (STD DDP)	9-2
diagnostic testing (16C452)	10-31
DICM - DMA Initialize Command register	8-5
digital I/O	
connector (J6)	3-22, 11-2, 13-4, 13-14, B-13
16C49 port	1-11
expansion module interface	12-1
interface characteristics	B-2
dimensions of the ZT 8802	B-7
direct memory access (DMA)	1-8, 3-12
advantages	3-12
DCU function	4-25
definition	E-3
operation	3-13

divisor latch access bit (DLAB)	10-14
DMA controller (V40) (see DCU - DMA Control Unit)	8-1
DMD - DMA Mode register	8-10
DMK - DMA Mask register	8-12
DOS	1-3, 2-6, 3-9, 3-24
cable requirements	2-13
configuring the ZT 8802	2-12
default jumper configuration	2-14, A-17
512K ROM/1M RAM configuration	A-18
memory maps	2-19, 2-20
memory requirements	2-12
powering up	2-17
with VGA board	3-24
DST - DMA Status register	8-11
DULA - DMA Unit Low Address register	5-6

- E -

E connector (STD 32 bus interface)	B-9
electrical specifications	3-21, 13-9, B-2
emulation mode (8080)	4-35
registers and flags	4-38
environmental specifications	2-5, B-2
EOI (end of interrupt), definition	E-3
EPROM	1-7, 2-9, 2-19, A-5, A-6
definition	E-3
establishing serial communications	10-4
event sense	
bit assignments	3-20
inputs (illustration)	13-12
interface (16C49)	1-11
operation	3-19, 13-11
expansion module	12-1
connector (J8)	12-4, B-15
external bus masters	4-3

- F -

features of the ZT 8802	1-4
Flash memory	A-5, A-15
definition	E-3
framing error (FE) indicator	10-25
frontplane	
definition	E-4
interrupts	A-7
functional blocks, ZT 8802	1-7
fuse	3-22

- G -

getting started	2-1
glossary	E-1
ground - jumper configuration	A-13

- H -

halt with interrupt restart (ZT 88CT02)	3-38
---	------

- I -

IBM PC compatibility	1-2, 3-8
ICU - Interrupt Control Unit	4-25, 7-2
definition	E-4
programmable registers	7-6
IER - Interrupt Enable register	10-29
IIR - Interrupt ID register	10-27
industrial I/O mounting racks	3-22, 13-14
inputs	
event sense	13-12
function	3-17, 13-8
installing the ZT 8802	2-6
interrupt controller	7-1
initialization words (IIW1-IIW4)	7-7
operation words (IMKW, IPFW, IMDW)	7-11

status words (IRQ, IIS, IPOL)	7-15
interrupts	1-10, 3-24
connector (J1)	B-11
jumper configuration	A-7
non-maskable	4-34
PIC interrupt request assignments	3-24
pin assignments	3-24
processing	4-31
vectors	3-28, 4-32
INTRQ*	1-11, 3-9, 3-26
definition	E-4
INTRQ1*, INTRQ2* (CNTRL*), INTRQ3*	1-11
definitions	E-4
I/O	
addressing	1-7, 2-30, 4-27
circuit operation	3-16
expansion module interface	12-1
line specifications	3-21
mapping	2-31, 4-27
module mounting racks	3-22, 13-14
points	3-15, 13-7
points, jumper options	3-33
point specifications	13-9
port assignments	10-15
port selection, 16C49 PIA	13-4
transfers	3-10
IOEXP	3-11
definition	E-4
Iol	3-21, 13-9
IULA - Interrupt Unit Low Address register	5-6

- J -

J1 interrupt inputs connector	B-11
J2 counter/timer connector	6-4, A-13, B-12
J3 RS-232-C interface connector	2-13, B-13

Index

J4 RS-232-C interface connector	2-13, B-13
J5 (reserved)	B-13
J6 digital I/O connector	3-22, 11-2, 13-4, 13-14, B-13
J7 RS-232-C DCE connector	2-13, 9-3, B-15
J8 SBX expansion module connector	12-4, B-15
J9 (reserved)	B-15
jumpers (see also W1, W2, etc.)	
customer configuration	A-16
description	A-1
DOS default configuration	2-15, A-17
DOS 512K ROM/1M RAM configuration	A-18
functional cross reference	A-2
STD ROM configuration	2-8, A-19

- K -

keyboard errors – effect on Counter/Timer 2 frequency	6-4
---	-----

- L -

latchup, preventing	C-1
LCR - Line Control register	10-19
LED	3-31, 3-37, 13-5
level-triggered interrupts	3-27
LSR - Line Status register	10-25
LSTTL, definition	E-4

- M -

mark (definition)	E-4
masking operation	3-18
mask port	3-18
mask register	3-18
MCR - Modem Control register	10-29
mechanical specifications	B-6
MEMEX (BHE*)	3-11
definition	E-4

jumper configuration	A-12
memory	
addressing	1-7, 2-19, 4-27
battery backup	A-3
Flash configuration	A-5
local memory	A-3
mode selection	2-28, 3-31, 13-5
RAM configuration	A-4
ROM configuration	A-5
transfers	3-10
mounting racks, I/O module	3-22, 13-14
MSR - Modem Status register	10-32

- N -

NMI (Non-Maskable Interrupt)	4-34
definition	E-4
NMIRQ* (Non-Maskable Interrupt Request)	3-30
definition	E-4

- O -

OPCN - On-Chip Peripheral Connector register	5-4
operating temperature	B-2
OPHA - On-Chip Peripheral High Address register	5-6
OPSEL - On-Chip Peripheral Selection register	5-5
Opto 22 I/O module mounting racks	13-14
oscillator disabling	3-31, 13-5
output	
function	3-16, 13-8
low voltage	3-21, 13-9
sink current	3-21, 13-9
output 1 (OUT1)	10-30
output 2 (OUT2)	10-30
overrun error (OE) indicator	10-25

- P -

parallel interface adapter (see PIA)	13-2
parallel I/O (see digital I/O)	1-11
parity error (PE) indicator	10-25
PBRESET*	3-31
PC-assisted (PCA) DOS system	2-13
definition	E-5
P connector (STD-80 bus interface)	B-9
peripherals	1-3
physical requirements of ZT 8802	2-4
PIA	3-15, 13-2, C-1
block diagram	13-2
Centronics printer interface	13-18
definition	E-5
digital I/O connector (J6)	3-22, 11-2, 13-4, 13-14, B-13
event sense port	13-11
features	13-3
interface, control port configuration	A-10
I/O circuit	13-7
I/O port	13-4, 13-10
I/O specifications	13-9
mask port	13-4, 13-12
ports	13-4
processor interface	13-4
reset	13-9
system setup considerations	C-1
polled interrupts on the STD bus	3-26
pop (stack operation)	4-34
definition	E-5
power	
conservation	1-11
consumption	1-2
fuse	3-22
requirements, ZT 8802	2-4
supply lines	3-22

power-fail protection	1-9, 3-34
power-up operation	3-21
prefetch	4-6, 4-9
definition	E-5
printer interface	13-18
80188 processor	3-5
processor configuration	5-2
programmable interrupt controller (PIC)	1-10, 3-24
definition	E-5
interrupt input requests	3-25
push (stack operation)	4-33, 4-36
definition	E-5

- R -

racks, I/O module	3-22, 13-14
RAM	
configuration	2-8, A-4
DOS default configuration (illustration)	2-15
socket, conversion to Flash	A-15
RD*, definition	E-5
readback	3-16, 13-8
real-time clock/calendar (DS 1202)	1-9, 14-1
byte mode vs. burst mode	14-7
registers	14-9
reset	3-31, 13-5
receive line signal detect (RLSD*)	10-11, 10-33
reliability	D-3
request to send (RTS*)	10-13, 10-30
RESERVED	1-11
reset	
effect on PIA I/O circuits	3-21
RESET* input signal	10-12
SYSRESET*, PBRESET* signals	3-31
V40 processor	4-26
return for service	D-4

Index

revision history	D-2
RFC - Refresh Control register	5-11
ring indicator (RI*)	10-33
inputs	10-11
RMA	D-4
ROM configuration	A-5
RS-232-C serial interface	3-23, 10-7
COM1 connector (J4)	2-13, B-13
COM2 connector (J3)	2-13, B-13
DCE connector (J7)	2-13, 9-3, B-15
definition	E-5
operation	9-3

- S -

SBX expansion module	12-1
connector (J8)	12-4, B-15
Schottky TTL, definition	E-6
SCM - Serial Command register	9-6
SCR - Scratchpad register	10-18
SCU - Serial Control Unit	2-4, 3-23, 4-24, 9-1
baud rate	9-9
definition	E-6
programmable registers	9-4
segment registers (V40 processor)	4-6
serial communications (16C452)	1-9, 2-4, 3-23, 10-1
interface	10-7
interrupt outputs (INT)	10-10
programmable registers	10-14
protocol	10-3
serial controller frequency	A-12
serial data inputs (SIN)	10-13
serial data outputs (SOUT)	10-13
signal definitions	10-9
serial communications (V40) (see SCU)	1-9, 9-1
silicon-controlled rectifier (SCR)	C-1
definition	E-6

silkscreen of ZT 8802 PCB	B-21
SIMK - Serial Interrupt Mask register	9-8
SMD - Serial Mode register	9-7
software support	1-3
development system	2-6
space (definition)	E-6
specifications	
cables	B-17
connectors	B-9
electrical and environmental	B-2
mechanical	B-6
SST - Serial Status register	9-5
standby mode	1-2
status indicator (LED)	3-31, 3-37, 13-5
STD bus	
cascaded interrupts	3-29
compatibility	1-11, 3-9
interface	3-10
interrupts	3-24, A-7
loading characteristics	B-3
polled interrupts	3-26
vectored interrupts	3-28
STD DDP (Device Driver Package)	9-2
STD ROM	1-3
cable requirements	2-10
configuring the ZT 8802	2-9
default jumper configuration	2-8, A-19
development system	2-6
jumper configuration	2-10
memory maps	2-19
memory requirements	2-9
software installation and operation	2-10
storage temperature	B-2
SULA - Serial Unit Low Address register	5-6
SYSRESET*	3-31
system requirements	2-4

Index

system setup considerations	C-1
systick timer	3-24

- T -

TCKS - Timer Clock Selection register	5-12
TCU - Counter/Timer Control Unit	4-25, 6-1
definition	E-6
I/O port addressing	6-5
programmable registers	6-5
technical assistance	D-2
theory of operation	3-2
timer clock	A-13
TMD - Timer Mode register	6-6
trailing edge of ring indicator (TERI)	10-32
transmit and receive buffer registers	10-18
transmitter holding register empty (THRE)	
indicator	10-26
interrupt	10-29
transmitter shift register empty (TEMT) indicator	10-26
TTL, definition	E-6
TULA - Timer/Counter Unit Low Address register	5-6
typical I/O circuit	3-15, 13-7

- U -

UART, definition	E-6
------------------------	-----

- V -

vectored interrupts	3-28
Vol	3-21, 13-9
V40 processor	1-2, 1-7, 3-5, 4-2
block diagram	4-5
configuration	5-2
functional blocks	4-4
interrupts	4-30

memory map	4-27
reset	5-14
VCR - V40 configuration registers	4-24, 5-2, E-6

- W -

W1, 28-29, 32-33 (battery backup)	A-3
W2-3 (timer clock selection)	A-13
W4, 22-25, 45-46 (control port)	A-10
W5-18, 47-51, 53-59 (STD bus interrupts)	A-7
W19-20, 30-31, 34-35 (RAM configuration)	A-4
W21, 40-44 (ROM configuration)	A-5
W26-27 (serial controller frequency)	A-12
W36-37 (factory options)	A-15
W38-39 (ROM device size selection)	A-6
W52 (MEMEX [BHE*])	A-12
W60 (digital ground and AUX GND)	A-13
wait-state generator	1-8, 5-8
warranty	D-5
watchdog timer	3-31, 11-1, 13-5
functional diagram	11-2
watchdog strobe interval	11-3, A-14
WCU - Wait Control Unit	4-24
definition	E-6
WCY1 - Wait Cycle 1 register	5-8
WCY2 - Wait Cycle 2 register	5-8
Western Digital 8250 serial interface	10-7
what's in the box	2-3
WMB - Wait Memory Boundary register	5-10
WR*, definition	E-7

- X -

XTP, definition	E-7
-----------------------	-----

- Z -

Ziatech DOS (see DOS)	1-3
ZT 2223 Industrial I/O Adapter Board	3-22, 13-14
ZT 2225 Industrial I/O Cable Adapter	3-22, 13-14, B-13
ZT 2226 24-Channel I/O Mounting Rack	3-22, 13-14, B-13
ZT 8802 vs ZT 8801	3-4
ZT 8802 vs ZT 8901	3-4
ZT 8950 Integrated Disk Drive/DMA Controller	3-24
ZT 8980 VGA card	3-9, 3-24
ZT 90022 digital I/O cable	13-14
ZT 90069 serial cable	2-10, 2-13, 3-23, B-17
ZT 90072 digital I/O cable	13-14
ZT 90089 digital I/O cable	13-14, B-13, B-18
ZT 90136 serial cable	2-13, B-19
ZT 90137 digital I/O cable	13-14
ZT 90156 printer interface cable	13-18, B-20
ZT 88CT02	1-2, 2-3, 3-37
added features	3-37
clock shutdown	3-38
halt with interrupt restart	3-38